

Universidade do Minho  
Escola de Engenharia

Rui Filipe Oliveira Gonçalves Vilaça

Estudo e desenvolvimento de um sistema de  
autolocalização para um veículo autónomo

Dissertação de Mestrado  
Ciclo de Estudos Integrados Conducentes ao  
Grau de Mestre em Engenharia Eletrónica Industrial e  
Computadores

Trabalho efetuado sob a orientação de  
Professor Doutor João Miguel Clemente de Sena Esteves  
Engenheiro Carlos Alberto Fernandes Arantes

# DECLARAÇÃO

**Autor:** Rui Filipe Oliveira Gonçalves Vilaça

**Correio eletrónico:** a71431@alunos.uminho.pt

**Número do cartão de cidadão:** 14756247 3ZY4

**Título da dissertação:** Estudo e desenvolvimento de um sistema de autolocalização para um veículo autónomo

**Ano de conclusão:** 2018

**Orientador:** Professor Doutor João Miguel Clemente de Sena Esteves

**Orientador CEiiA:** Engenheiro Carlos Alberto Fernandes Arantes

**Designação do Mestrado:** Ciclo de Estudos Integrados Conducentes ao Grau de Mestre em Engenharia Eletrónica Industrial e Computadores

Departamento de Eletrónica Industrial

Escola de Engenharia

Universidade do Minho

DE ACORDO COM A LEGISLAÇÃO EM VIGOR, NÃO É PERMITIDA A REPRODUÇÃO DE QUALQUER PARTE DESTA DISSERTAÇÃO.

Guimarães, \_\_\_\_ / \_\_\_\_ / \_\_\_\_

Assinatura: \_\_\_\_\_

# Agradecimentos

A conclusão desta dissertação resume 5 anos de esforço e dedicação. Deste modo, gostaria de deixar os meus sinceros agradecimentos a todos que me acompanharam durante este percurso.

Quero agradecer ao Professor Doutor João Sena Esteves e ao Engenheiro Carlos Arantes por todo o apoio e disponibilidade durante a realização desta dissertação.

Ao CEiiA pela oportunidade de realizar a dissertação nas suas instalações e a toda a equipa de Mobilidade pelo acompanhamento dado, com um especial obrigado ao Engenheiro Tiago Macedo, Doutor João Figueiredo e ao Nuno Carneiro.

Aos meus amigos/parceiros Francisco Nunes, Jorge Ramos e Ricardo Martins por todos os momentos passados juntos durante estes 5 anos.

Por fim, mas não por último, aos meus pais por todo o incentivo e sacrifício que fizeram para me proporcionar estes 5 anos de formação académica superior. Ao meu irmão e à minha cunhada, por todas as conversas e conselhos dados e claro, à minha afilhada pela capacidade de me conseguir distrair de todos os problemas. Deixo ainda aqui um enorme agradecimento à minha namorada por toda a paciência, conselhos, conversas, apoio e incentivo ao longo destes muitos anos.

Muito Obrigado!



# Resumo

O desenvolvimento de sistemas de localização robustos, eficientes e de baixo custo permanece como uma das questões importantes dos veículos autónomos.

Esta dissertação foi desenvolvida em parceria com o CEiiA e tem como principal objetivo o desenvolvimento de um sistema de autolocalização de baixo custo para um veículo autónomo que circulará no exterior. O cumprimento deste objetivo envolveu a especificação, implementação e teste de um protótipo. Envolveu também o desenvolvimento do *firmware* do microcontrolador (C/C++) para realizar a recolha e comunicação dos dados recolhidos pelos diversos sensores. A plataforma móvel utilizada para testes foi uma bicicleta na qual se integraram um módulo GPS, um *encoder* absoluto, um alternador (utilizado como um *encoder* incremental), um giroscópio, um acelerómetro e um magnetómetro. Os dados são adquiridos a uma frequência de 40 Hz e guardados num cartão  $\mu SD$  ou transmitidos através de um módulo de comunicação sem fios *XBee*. Para realizar a fusão sensorial exploraram-se o filtro complementar, o filtro de *Kalman Extended* e o filtro de *Kalman Unscented*, através do *software Matlab*. De forma a permitir a visualização dos resultados, desenvolveu-se uma interface gráfica em HTML, CSS e *Javascript*.

Realizaram-se diversos testes ao sistema, tendo-se obtido na medição da posição do veículo um erro máximo de 10 m por parte do GPS. Foi também verificado que o erro da odometria é proporcional ao aumento da complexidade da trajetória e da distância percorrida. De entre os três filtros aplicados, o filtro complementar revelou ser uma solução satisfatória para percursos simples. Não houve diferenças significativas entre os resultados obtidos com os diferentes filtros de *Kalman*.

**Palavras-chave:** Sistemas de Localização, Veículos Autónomos, Autolocalização, Fusão Sensorial.



# Abstract

The development of robust, efficient and low-cost localization systems remains one of the important issues of autonomous vehicles.

This dissertation was developed in partnership with CEiiA and its main objective is the development of a low-cost self-localization system for an autonomous vehicle that will circulate in a non-controlled environment. The fulfillment of this objective involved the specification, implementation and testing of a prototype. It also involved the development of microcontroller firmware (C/C++) to gather and communicate the data collected by the various sensors. The mobile platform used for testing was a bicycle in which a GPS module, an absolute encoder, an alternator (used as an incremental encoder), a gyroscope, an accelerometer and a magnetometer were integrated. The data is acquired at a frequency of 40Hz and stored on a  $\mu$ SD card or transmitted through an XBee wireless communication module. The Complementary filter, the Extended Kalman filter and the Unscented Kalman filter were explored through Matlab software. In order to allow the visualization of the results, a graphical user interface was developed in HTML, CSS and Javascript.

Several tests were performed on the system, and a maximum error of 10 m was obtained by the GPS in the vehicle position measurement. It was also verified that the error of the odometry is proportional to the increase of the complexity and distance covered. Of the three filters applied, the complimentary filter proved to be a satisfactory solution for simple routes. There was no significant differences between the results obtained with the different Kalman filters.

**Keywords:** Localization Systems, Autonomous Vehicles, Self-Localization, Sensory Fusion.





# Índice

Agradecimentos.....	iii
Resumo.....	v
Abstract.....	vii
Índice.....	ix
Índice de figuras.....	xi
Índice de tabelas.....	xiii
Lista de siglas e acrónimos.....	xv
Nomenclatura.....	xvii
1. Introdução.....	1
1.1. Enquadramento.....	1
1.2. CEiiA.....	6
1.3. Motivação.....	6
1.4. Objetivos.....	7
1.5. Proposta do trabalho a desenvolver.....	7
1.6. Organização da dissertação.....	8
2. Estado de Arte.....	9
2.1. GPS.....	9
2.2. Utilização conjunta do GPS com INS.....	10
2.3. Utilização conjunta de GPS, INS e Odometria.....	12
3. Fundamentos Teóricos.....	15
3.1. Métodos de localização.....	15
3.1.1. Localização baseada em trilateração com balizas.....	15
3.1.2. Utilização de sensores inerciais.....	17
3.1.3. Odometria.....	17
3.2. Representação de Orientação.....	19
3.2.1. Matriz de Rotação.....	19
3.2.2. Ângulos de Euler.....	20
3.2.3. Vetor de Rotação.....	22
3.2.4. Quaterniões.....	23
3.3. Filtros Matemáticos.....	24
3.3.1. Filtro Complementar.....	25

## Índice

---

3.3.2.	Filtro de Kalman .....	25
3.3.3.	Filtro de Kalman <i>Extended</i> .....	28
3.3.4.	Filtro de Kalman <i>Unscented</i> .....	28
3.4.	Protocolos de comunicação de baixo nível .....	32
3.4.1.	UART .....	32
3.4.2.	I2C .....	33
3.4.3.	SPI .....	34
4.	Arquitetura do sistema de localização .....	37
4.1.	Unidade de Recolha de Dados .....	38
4.2.	Unidade de Processamento de Dados .....	38
4.3.	Interface Gráfica .....	39
5.	Protótipo do sistema de localização .....	41
5.1.	Unidade de Recolha de Dados .....	41
5.1.1.	GPS .....	45
5.1.2.	IMU .....	47
5.1.3.	Alternador .....	48
5.1.4.	Encoder Absoluto .....	50
5.1.5.	Comunicação de Dados .....	52
5.1.6.	Módulo de Alimentação .....	54
5.2.	Cinemática do veículo .....	57
5.3.	Unidade de Processamento de Dados .....	59
5.3.1.	Filtro Complementar .....	61
5.3.2.	Filtro de <i>Kalman Extended</i> .....	64
5.3.3.	Filtro de <i>Kalman Unscented</i> .....	67
5.4.	Interface Gráfica .....	68
6.	Resultados Experimentais .....	71
6.1.	Teste 1 .....	71
6.2.	Teste 2 .....	73
6.3.	Teste 3 .....	74
6.4.	Análise comparativa dos testes .....	75
7.	Conclusões e trabalho futuro .....	77
	Referências .....	79
	Apêndices .....	87
	Apêndice A .....	87

---

# Índice de figuras

Figura 1 - (a) <i>NovAtel</i> PwrPak7-E1 [17] e (b) <i>XSENS</i> MTi-G-710 [18].....	4
Figura 2 – a) AGV <i>Kiva</i> utilizado nos armazéns da <i>Amazon</i> em funções de transporte de mercadoria; b) Uma das câmaras do AGV <i>Kiva</i> [20]. ....	5
Figura 3 - <i>Toyota Prius</i> modificado pela <i>Google</i> [16]. ....	5
Figura 4 - Esquema geral do sistema a desenvolver. ....	8
Figura 5 – Diferentes situações passíveis de acontecer durante a propagação do sinal GPS. ....	9
Figura 6 – Diagrama temporal do sistema de localização baseado na utilização conjunta do GPS e do INS. ....	10
Figura 7 - (a) Odin, veículo usado pela equipa VitorTango's no DARPA Urban Challenge (adaptado de [27]) e (b) ATRV-Jr desenvolvido por Panzieri <i>et. al.</i> [28].....	11
Figura 8 - <i>Encoder</i> Absoluto EMS22A [34]. ....	12
Figura 9 – Diagrama temporal do sistema de localização baseado na utilização conjunta do GPS, INS e odometria. ....	13
Figura 10 - Veículo adaptado pela Universidade de Tsinghua [33]. ....	13
Figura 11 - Localização usando uma baliza. ....	16
Figura 12 - Localização usando duas balizas. ....	16
Figura 13 - Localização usando três balizas. ....	16
Figura 14 - Representação do referencial global e do referencial de corpo. ....	19
Figura 15 - Referencial $\{B\}$ obtido através da rotação sequencial dos três eixos do referencial $\{A\}$ . ....	20
Figura 16 - Fenómeno <i>Gimbal Lock</i> (Adaptado de [39]) .....	22
Figura 17 - Rotação de um referencial $\{A\}$ através de um vetor de rotação $\lambda$ para obter um referencial $\{B\}$ . ....	22
Figura 18 - Rotação de um referencial $\{A\}$ através de um eixo de rotação $\mathbf{q}$ e um ângulo $\theta$ obtendo o referencial $\{B\}$ . ....	24
Figura 19 – Ciclo do filtro de Kalman. ....	26
Figura 20 - Visão concetual do filtro de Kalman.....	27
Figura 21 - Transformação das fórmulas matemáticas do filtro de Kalman para o filtro de Kalman Extendido. ....	28
Figura 22 – Esquema de uma comunicação UART.....	33
Figura 23 - Esquema de um barramento I2C. ....	34
Figura 24 - Esquema de uma conexão SPI [58]. ....	35
Figura 25 - Arquitetura geral do sistema implementado, evidenciando os diferentes níveis de tensão, os sensores utilizados e protocolos de comunicação.....	37
Figura 26 - Arquitetura geral da unidade de processamento de dados.....	39
Figura 27 - Arquitetura geral da interface gráfica desenvolvida. ....	39
Figura 28 - a) Parte do esquemático desenhado; b) <i>Layout</i> superior da PCI desenvolvida .....	42
Figura 29 - a) Face inferior da PCI produzida; b) Face superior da PCI produzida.....	42
Figura 30 - Fluxograma do código desenvolvido para a unidade de recolha de dados. ....	44
Figura 31 - a) Desenho em CAD das peças para acoplar a URD à bicicleta; b) Peças acopladas à bicicleta utilizada para testes. ....	44
Figura 32 - Recetor GPS Quectel L80 M39 [64]. ....	45
Figura 33 - Exemplo de uma <i>string</i> NMEA 0183 por descodificar, retirada do recetor utilizado. ....	46
Figura 34 - a) Level shifter unidirecional; b) Level shifter bidirecional (adaptado de [67]). ....	46
Figura 35 - <i>Adafruit BNO055 9-DOF</i> [68]. ....	47

## Índice de figuras

---

Figura 36 - Esquema de um alternador com dois pólos e a onda sinusoidal gerada por este [71].	48
Figura 37 - Alternador Shimano Nexus DH-C6000 utilizado no projeto [72].	49
Figura 38 - Circuito eletrónico desenvolvido para transformar o sinal alternado num sinal contínuo.	49
Figura 39 - a) Encoder Bourns EMS22A utilizado; b) esquema do funcionamento do encoder na unidade de recolha de dados.	50
Figura 40 - Protocolo para ler o valor do encoder absoluto <i>Bourns EMS22A</i> [34].	51
Figura 41 - Topologias de rede possíveis de construir utilizando o módulo <i>XBee Pro S2B</i> [35].	53
Figura 42 - Exemplo de um excerto de uma comunicação sem fios e de um ficheiro .csv.	54
Figura 43 - Arquitetura do sistema de alimentação implementado na URD.	57
Figura 44 - Bicicleta utilizada para validar o protótipo do sistema de autolocalização desenvolvido.	57
Figura 45 - Modelo cinemático do automóvel e modelo da bicicleta. As rodas do automóvel estão representadas a cinzento claro e as rodas da bicicleta a cinzento escuro.	58
Figura 46 - Algoritmo geral da UPD.	59
Figura 47 - Função implementada da compensação de inclinação do magnetómetro.	60
Figura 48 - Fluxograma referente à função Filtro Complementar.	62
Figura 49 - Fluxograma referente à função <i>Kalman_Filter_Extended</i> .	64
Figura 50 - Fluxograma referente à função <i>Unscented_Kalman_Filter</i> .	67
Figura 51 - Interface gráfica do sistema de localização.	69
Figura 52 - a) Percurso realizado no Teste 1 evidenciando a distância, ponto de partida e chegada. b) Diferentes trajetórias medidas e calculadas pelos diferentes sensores e filtros, respetivamente.	71
Figura 53 – Aumento da parte final da figura 53.b).	72
Figura 54 - Orientação do veículo durante o Teste 1.	72
Figura 55 - a) Percurso realizado no Teste 2 evidenciando a distância, ponto de partida e chegada. b) Diferentes trajetórias medidas e calculadas pelos diferentes sensores e filtros, respetivamente.	73
Figura 56 - Orientação do veículo durante o teste 2.	74
Figura 57 - a) Percurso realizado no Teste3 evidenciando a distância, ponto de partida e ponto de chegada; b) Diferentes trajetórias medidas e calculadas pelos diferentes sensores e filtros, respetivamente.	75
Figura 58 - Orientação do veículo durante o Teste 3.	75

# Índice de tabelas

Tabela 1 - Comparação entre os diferentes filtros matemáticos (adaptado de [57]). .....	32
Tabela 2 - Diferenças entre conversores eletrónicos de potência lineares e comutados (adaptado de [35]).....	55
Tabela 3 - Consumo dos componentes utilizados na URD. ....	56
Tabela 4 - Resumo dos testes realizados. ....	76



# Lista de siglas e acrónimos

Abreviatura	Designação completa
AGV	<i>Automated Guided Vehicle</i>
API	<i>Application Programming Interface</i>
BPS	<i>Bits Per Second</i>
CAD	<i>Computer Aided Design</i>
CEP	<i>Circular Error Probable</i>
CPU	<i>Central Processing Unit</i>
CSS	<i>Cascading Style Sheets</i>
DARPA	<i>Defense Advanced Research Projects Agency</i>
DCM	<i>Direct Cosine Matrix</i>
DOF	<i>Degrees of Freedom</i>
EUA	Estados Unidos da América
FKE	Filtro de Kalman Extended
FKU	Filtro de Kalman <i>Unscented</i>
GPS	<i>Global Positioning System</i>
HDOP	<i>Horizontal Dilution of Precision</i>
HTML	<i>HyperText Markup Language</i>
I2C	<i>Inter-Integrated Circuit</i>
I&D	Investigação e Desenvolvimento
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IMU	<i>Inertial Measurement Unit</i>
INS	<i>Inertial Navigation System</i>
MEMS	<i>Micro-Electro-Mechanical Systems</i>
MIMO	<i>Multi-Input/Multi-Output</i>
MISO	<i>Multi-Input/Single-Output</i>
MISO	<i>Master-In/Slave-Out</i>
MOSI	<i>Master-Out/Slave-In</i>
NMEA	<i>National Marine Electronics Association</i>
PCI	Placa de Circuito Impresso
QR	<i>Quick Response</i>
RoHs	<i>Restriction of Certain Hazardous Substances</i>
SCK	<i>Synchronous Clock</i>
SPI	<i>Serial Peripheral Interface</i>
SS	<i>Slave-Select</i>



### Lista de siglas e acrónimos

<b>Abreviatura</b>	<b>Designação completa</b>
TOA	<i>Time of Arrival</i>
TWI	<i>Two Wire Interface</i>
UART	<i>Universal Asynchronous Receiver/Transmitter</i>
UPD	Unidade de Processamento de Dados
URD	Unidade de Recolha de Dados
WWW	<i>World Wide Web</i>

# Nomenclatura

$\hat{x}_k^-$	Estado predito num dado instante k
$\hat{x}_{k-1}$	Último estado
$A$	Modelo de estados linear
$B$	Matriz de sinais de controlo
$u_{k-1}$	Entrada de controlo
$P_k$	Matriz de covariância
$Q$	Matriz de covariância do erro do modelo do sistema
$K_k$	Ganho de Kalman
$H$	Matriz de observação
$R$	Matriz de covariância do erro das medições
$W_M$	Pesos das amostras da média
$W_C$	Pesos das amostras da covariância
$\chi_{m_k}$	Matriz das amostras
$\chi_{m_{k-1}}$	Matriz das amostras propagadas pelo modelo e sinais de controlo
$\psi_m$	Matriz das amostras propagadas pelas medições
$P_{yy}$	Matriz de covariância da saída predita
$P_{xy}$	Matriz de covariância entre o estado e a saída
$D_{pp}$	Distância por pulso
$\eta$	Rendimento
$P_{OUT}$	Potência de saída
$P_{IN}$	Potência de entrada
$V_{OUT}$	Tensão de saída
$V_{IN}$	Tensão de entrada
$I$	Corrente
$\dot{x}$	Velocidade linear relativamente ao eixo do -x
$\dot{y}$	Velocidade linear relativamente ao eixo do -y
$\dot{\theta}$	Velocidade angular relativamente ao eixo do -z
$ v $	Módulo da velocidade linear

## Nomenclatura

---

$\theta$	Orientação do veículo
$\gamma$	Ângulo da direção do veículo
$L$	Distância entre as rodas dianteiras e traseiras do veículo
$Roll$	Ângulo relativo ao eixo do -x
$Pitch$	Ângulo relativo ao eixo do -y
$Yaw$	Ângulo relativo ao eixo do -z
$Fused_{yaw}$	Ângulo calculado apenas com IMU
$R_{yaw}$	Ângulo cálculo apenas com a odometria e ângulo de direção do veículo
$P_A$	Número de pulsos gerados pelo alternador num intervalo de tempo
$x_{rel}$	Posição relativa ao eixo -x calculada apenas com a odometria e o ângulo de direção do veículo
$y_{rel}$	Posição relativa ao eixo -y calculada apenas com a odometria e o ângulo de direção do veículo
$x$	Posição do veículo em graus (Latitude)
$y$	Posição do veículo em graus (Longitude)
$v$	Velocidade linear do veículo
$\omega$	Velocidade angular do veículo

# 1. Introdução

Neste capítulo introdutório serão apresentados o enquadramento do trabalho, a definição de localização, a motivação, os objetivos, alguns fundamentos teóricos, bem como a proposta de trabalho a desenvolver. Por último, será também descrita a organização do documento.

## 1.1. Enquadramento

Os veículos autónomos são um tema cada vez mais recorrente no dia a dia da sociedade. Uma das características destes veículos diz respeito à capacidade de se deslocarem, eficientemente, de um ponto A até a um ponto B, sem a intervenção humana. Para tal, necessitam de capacidades para determinarem a sua posição e orientação [1][2]. De acordo com Leonard, J. J. e Durrant-Whyte, H. F. [3] o problema da navegação autónoma pode ser dividido em três grandes questões: “*onde estou?*”, “*para onde vou?*” e “*como fazer para lá chegar?*”. De entre estas questões, apenas a primeira se relaciona com a temática da localização.

Existe alguma discórdia entre autores quanto à definição de **localização**. Leonard e Durrant-White, citados em [4], definem *localização* como o “*processo de determinar a posição de um robô num referencial global utilizando a informação proveniente de sensores*”. Esta definição, aplicável a veículos em geral e não apenas a robôs, é vista como incompleta por autores, como J. Borenstein em [5] e J. Sena Esteves em [4], que defendem que a localização engloba não só a determinação da posição do veículo, mas também a determinação da sua orientação. Neste trabalho será adotada esta última definição. A **autolocalização** consiste na capacidade de um veículo determinar autonomamente a sua posição e orientação relativamente a um referencial. Os métodos de localização podem dividir-se em dois grupos: métodos de localização absoluta e métodos de localização relativa [4]. A localização absoluta refere-se à “*possibilidade de um robô móvel determinar a sua posição e orientação [...] de um modo independente de suposições sobre movimentos anteriores*” [4][6]. Esta definição é aplicável a veículos em geral e não apenas a robôs. Alguns métodos de localização absoluta são baseados em:

- **Utilização de balizas** – As balizas (*beacons*) podem ser definidas como dispositivos que, colocados numa posição previamente conhecida, permitem determinar de forma absoluta e contínua a posição (em alguns casos, também a orientação) de um veículo

enquanto este mantiver uma comunicação em linha de vista (*line of sight*) com uma ou mais balizas. A determinação da posição pode ser feita recorrendo a métodos, tais como, a trilateração, que tem por base a medição de distâncias [5], a triangulação, que tem por base a medição de ângulos e que também permite a determinação da orientação [7], ou outros métodos baseados na medição simultânea de ângulos e distâncias [4][3];

- **Reconhecimento de marcos naturais** – Os marcos naturais são referências inconfundíveis e inerentes ao meio com localizações previamente conhecidas e importantes. Podem ser formas geométricas simples (círculos ou retângulos) como portas ou janelas, luzes do tecto, entre outros. Uma vez que são naturais ao meio não existe necessidade de o preparar [4][8];
- **Reconhecimento de marcos artificiais** – Os marcos artificiais são referências distintas colocadas no meio, em localizações previamente definidas, conhecidas e importantes, tal como no caso anterior. Contudo, e sendo artificiais, são otimizados para o fácil reconhecimento. Desta forma, podem conter cores, códigos de barras ou códigos QR (*Quick Response*). Comparativamente aos marcos naturais, estes possibilitam a sua otimização para a deteção [4][9];
- **Correspondência de mapas** – Este método baseia-se na comparação de um mapa-modelo do ambiente, em que o sistema autónomo circulará, com os dados que os sensores instalados recolhem. Se existir uma correlação entre os dados recolhidos e os do modelo, então a posição do robô pode ser determinada. Normalmente estes mapas são geométricos (representam o mundo num sistema de coordenadas) ou topológicos (representam o mundo numa rede de arcos e nós) e necessitam de ser o mais exatos possível [4][10].

A localização relativa consiste em determinar a posição e a orientação de um veículo a partir de informações sobre movimentos anteriores. É frequente encontrar a designação de *dead reckoning* para métodos de localização relativa, que são baseados em [4][6]:

- **Odometria** – É um método que se encontra definido como a determinação da distância percorrida por um veículo, da sua orientação e da sua velocidade recorrendo

à medição da rotação das rodas. Este método não depende de referências externas, sendo capaz de fornecer informações de uma forma ininterrupta. Contudo, é um método que apenas se pode confiar nos dados por distâncias reduzidas devido à contínua acumulação de erros [4][8][5];

- **Navegação inercial** – É um método que se baseia no uso de sensores inerciais, como o giroscópio ou o acelerómetro, para medir a velocidade angular e aceleração linear, respetivamente. Através da integração destes valores é possível determinar a posição e orientação do veículo. Como no caso da odometria, é um método que não depende de referências externas e apenas pode ser usado por curtos períodos de tempo devido à contínua acumulação de erros [4][11][5];
- **Utilização de sensores *Doppler*** – Os sensores *Doppler* funcionam baseando-se no efeito *Doppler*, “*que consiste na mudança de frequência que se observa quando a energia radiada por um emissor é refletida por uma superfície que se move em relação a esse emissor*” [4]. Habitualmente, este tipo de sensores não é aplicado a veículos terrestres devido à complexidade de processamento de dados e ao custo destes. A localização baseado em sensores *Doppler* pode, no entanto, ser uma alternativa adequada ao uso da odometria [5].

Os métodos de localização usados de forma individual poderão ser incapazes de garantir uma exatidão satisfatória, em determinadas aplicações. A solução consiste normalmente na utilização conjunta de, pelo menos, um método de localização absoluta e um método de localização relativa. A título de exemplo, os sistemas INS/GPS (*Inertial Navigation System/Global Positioning System*), são sistemas de localização que funcionam através da fusão de dados provenientes do GPS e de sensores inerciais (giroscópios e/ou acelerómetros). Em situações onde o sinal GPS é de fraca qualidade ou até mesmo inexistente, os sensores inerciais oferecem uma solução económica, robusta e fiável de forma a complementar o GPS. Este tipo de sistemas de localização poderá, por vezes, ser complementado com sistemas de visão (câmaras), *encoders*, bússolas ou outras tecnologias, aumentando a sua complexidade e por conseguinte o seu custo [12].

Cada caso deverá ser estudado cuidadosamente devido à enorme variedade de sensores e tecnologias existentes para o efeito, que se podem aplicar em aviões, barcos, veículos agrícolas,

pedestres, automóveis, equipamento militar – armas e veículos, entre muitos outros [3], [13], [14], [15], [16]. Empresas, tais como, a *NovAte*/ou a *XSENS*, comercializam equipamentos que integram vários destes sensores numa única unidade. Na Figura 1.a) e 1.b) é possível ver dois exemplos, comercializados pela *NovAte*/e pela *XSENS*, respetivamente.



Figura 1 - (a) *NovAte*/PwrPak7-E1 [17] e (b) *XSENS* MTi-G-710 [18].

Os AGV's (*Automated Guided Vehicle*) e os automóveis autónomos da *Google* são dois exemplos distintos da aplicação destes sistemas de localização. Os AGV's mais modernos podem usar radares, câmaras, sensores inerciais, *encoders*, sensores ultrassónicos, entre outros. A título de exemplo, é possível visualizar na Figura 2 o AGV *Kiva*, utilizado nos armazéns da *Amazon* para transporte de mercadorias. Usa duas câmaras, uma direcionada para cima de forma a identificar qual a estante correta e outra direcionada para baixo de forma a determinar qual a rota em que se encontra, e sensores infravermelhos de forma a evitar colisões. Este tipo de sistema de localização é eficiente, encontrando-se, ainda, muito limitado, uma vez que os AGV's apenas podem circular no espaço que lhes é destinado. Os AGV's são frequentemente usados em tarefas de transporte, mas, podem ser programados para realizar outro tipo de tarefas [19][20].



Figura 2 – a) AGV *Kiva* utilizado nos armazéns da *Amazon* em funções de transporte de mercadoria; b) Uma das câmaras do AGV *Kiva* [20].

Nos automóveis autónomos o desafio é maior uma vez que o ambiente tem um maior grau de imprevisibilidade. No caso dos automóveis autónomos da *Google*, o sistema de localização é composto por um recetor GPS, um acelerómetro, um giroscópio e *encoders*. Na Figura 3 é possível ver um desses automóveis, um *Toyota Prius* modificado. O sistema de localização está preparado para funcionar em ambientes exteriores sem preparação prévia do meio, ao contrário do que sucede com o sistema de localização usado nos robôs *Kiva* da *Amazon*. Em contrapartida, uma vez que usa o GPS como principal sistema de localização, enfrenta importantes dificuldades em interiores [16].



Figura 3 - *Toyota Prius* modificado pela *Google* [16].

Os exemplos acima descritos, evidenciam a necessidade da escolha do sistema de localização que melhor se adequa a cada problema específico. Assim, e de forma a otimizar o



veículo autónomo à tarefa pretendida, o respetivo sistema de localização é desenvolvido de acordo com o contexto em que o veículo circulará. Esta otimização é feita tendo em conta o custo aceite e o desempenho pretendido para o sistema de localização. No caso do AGV *Kiva*, este é otimizado para funcionar unicamente dentro do armazém, já no caso do veículo da *Google* o sistema é otimizado para circular em exteriores.

### 1.2. CEiiA

O CEiiA é um **Centro de Engenharia e Desenvolvimento de Produto** que desenvolve e opera produtos e serviços nas indústrias de mobilidade, nomeadamente Automóvel e Mobilidade Urbana, Aeronáutica, Mar e Espaço estando presente em vários países da Europa e da América do Sul.

O CEiiA é um forte investidor de I&D (Investigação e Desenvolvimento) em Portugal, sendo atualmente uma referência internacional na área da mobilidade sustentável e reconhecido no mundo aeronáutico pelas suas competências em engenharia de estruturas. De forma a contribuir para a evolução económica e industrial em Portugal, tem liderado programas como o KC-390 da Embraer.

No Automóvel e Mobilidade, o CEiiA desenvolve novos conceitos de mobilidade que envolvem novos dispositivos, novos serviços de mobilidade e novos modelos de negócio mais sustentáveis para as cidades [21].

### 1.3. Motivação

A localização de veículos é uma área bastante estudada. De acordo com Leonard, J. J. e Durrant-Whyte, H. F., a dificuldade que lhe é inerente encontra-se na aquisição, interpretação e correlação de informação recolhida pelos diversos sensores utilizados, de uma forma fiável e robusta [3]. Algumas empresas já comercializam sistemas de localização (Figura 1) que nem sempre apresentam a exatidão e robustez suficiente para veículos, nomeadamente o tradicional automóvel, devido às velocidades e ambientes com que estes podem circular. Além disso, o preço praticado por essas empresas é, geralmente, ainda bastante elevado para a inclusão na produção em massa de veículos. Assim, o desenvolvimento de sistemas de autolocalização para veículos

autónomos que sejam *precisos*, *robustos* e de *baixo custo* permanece um desafio importante [2][3][22].

O CEiiA pretende inovar na área da mobilidade terrestre e, como tal, almeja desenvolver um protótipo de um veículo autónomo. A autolocalização do veículo é um dos muitos desafios inerentes ao projeto, sendo necessário o desenvolvimento de um sistema de autolocalização com uma adequada relação qualidade/preço.

### 1.4. Objetivos

O principal objetivo desta dissertação é o desenvolvimento de um protótipo de um sistema de autolocalização, de baixo custo, para um veículo autónomo. O sistema utiliza métodos de localização absoluta e localização relativa para determinar a posição e a orientação do veículo. De forma a se atingir o objetivo principal, alguns tarefas intermédias tiveram de ser cumpridas:

- Desenvolvimento de um sistema central de recolha de dados;
- Desenvolvimento de um sistema central de processamento de dados;
- Implementação de uma comunicação contínua e sem erros entre os diferentes sistemas;
- Implementação de um método matemático de fusão de dados;
- Desenvolvimento de um ambiente gráfico para a interação com o utilizador.

### 1.5. Proposta do trabalho a desenvolver

Propõe-se o desenvolvimento de um protótipo de um sistema de autolocalização baseado em fusão sensorial para um veículo autónomo. Será necessário o desenvolvimento do *hardware* necessário para a integração dos diferentes componentes do sistema, implementar os diferentes filtros/algoritmos matemáticos e desenvolver-se uma interface gráfica. Na Figura 4 está representado um esquema geral do sistema a desenvolver.

O sistema recolherá diversos dados provenientes de diversos sensores e reporta-los-á, através de uma comunicação com ou sem fios, para uma unidade de processamento. Nesta unidade de processamento, os dados serão tratados e fundidos de forma a determinar a posição, a orientação e a velocidade do veículo em questão. Para facilitar a interpretação dos resultados, estes serão graficamente transpostos para um mapa.

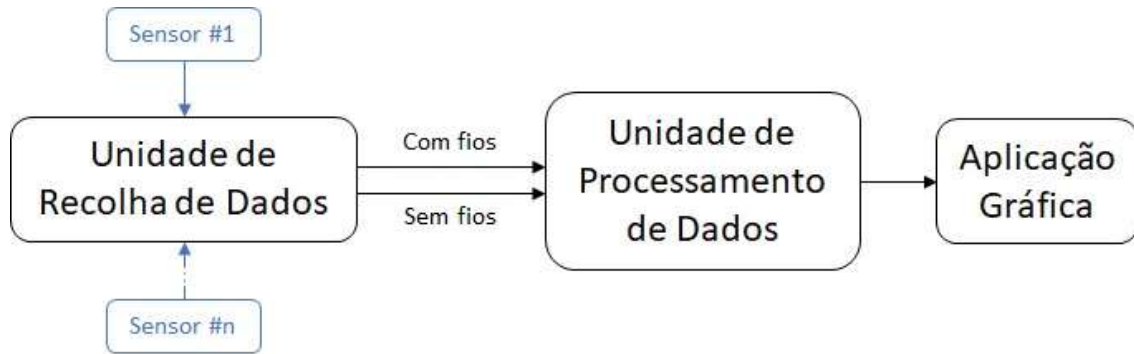


Figura 4 - Esquema geral do sistema a desenvolver.

### 1.6. Organização da dissertação

O presente documento encontra-se dividido em seis capítulos.

No capítulo 2 apresenta-se o estado de arte relativo à proposta de trabalho.

No capítulo 3 apresentam-se os fundamentos teóricos da dissertação, descrevendo-se os métodos de localização e representação de orientação que mais se adequam à superação do desafio.

No capítulo 4 descreve-se a arquitetura do sistema de localização.

No capítulo 5 descreve-se o trabalho realizado a nível de *hardware*, *firmware* e *software*.

No capítulo 6 apresentam-se os testes realizados ao sistema e a respetiva análise comparativa.

Por fim, no capítulo 7, expõem-se as conclusões gerais do trabalho e algumas sugestões de trabalho futuro.

## 2. Estado de Arte

Neste capítulo são apresentados sistemas de localização que recorrem a diferentes sensores e métodos para realizar a localização. Dependendo da aplicação, o sistema pode ser mais ou menos complexo.

### 2.1. GPS

O **GPS** pertence aos Estados Unidos da América (EUA) e é constituído por mais de 30 satélites. Este foi desenvolvido de forma a ser capaz de fornecer informações para um número ilimitado de utilizadores, uma vez que os recetores funcionam de forma passiva, ou seja, apenas recebem os sinais emitidos pelos satélites [5].

O GPS é capaz de determinar a posição de qualquer recetor à face da Terra desde que seja visível por pelo menos quatro satélites em simultâneo. O sistema baseia-se na receção de sinais satélite pelo conceito de *Time of Arrival* (TOA), enfrentando grandes dificuldades em interiores. Esta dificuldade reside na possibilidade de a receção do sinal ser distorcida ou até mesmo totalmente bloqueada pelos materiais que constituem os edifícios, ou sofrer reflexões de sinal, aumentando o tempo de propagação deste. Este efeito é conhecido como *multipath* e muitas vezes é impossível de detetar. A Figura 5 ilustra estes problemas [5][7].

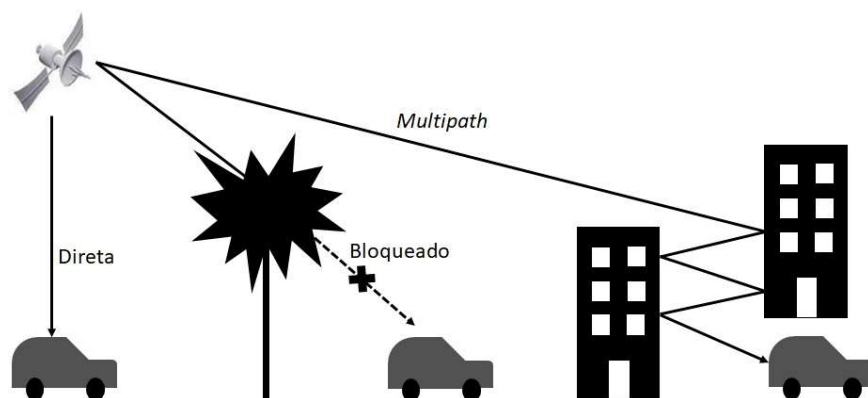


Figura 5 – Diferentes situações possíveis de acontecer durante a propagação do sinal GPS.

Todavia, este é até ao momento o melhor sistema de localização absoluta sendo utilizado em praticamente todos os sistemas de localização para veículos que circulam maioritariamente em exteriores. No entanto, quando usado como a única fonte de localização, pode não oferecer a exatidão e precisão necessária para determinadas aplicações.

## 2.2. Utilização conjunta do GPS com INS

A **utilização conjunta de sensores inerciais (INS) e GPS** é uma fusão sensorial comum devido à complementariedade mútua entre o GPS e os INS. Fusões deste género podem ser encontradas na literatura em [15], [23], [24] e [25]. Estes sistemas, de uma forma geral, recorrem ao GPS como referência do sistema e, entre leituras deste, a localização é feita recorrendo aos dados fornecidos pelos INS e a última leitura do GPS. Isto acontece porque a frequência de amostragem do GPS é tipicamente inferior à frequência de amostragem dos INS. A Figura 6 ilustra o funcionamento dos sistemas de localização deste género e encontra-se evidenciado quais as leituras utilizadas em cada instante de tempo.

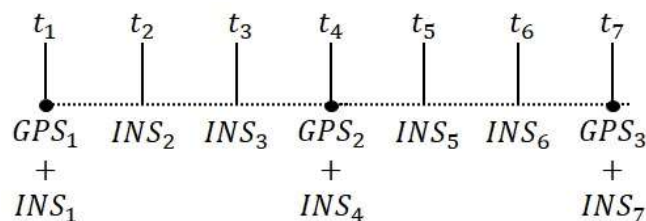


Figura 6 – Diagrama temporal do sistema de localização baseado na utilização conjunta do GPS e do INS.

Os sensores inerciais (giroscópio e acelerómetro) são denominados de INS quando usados para efeitos de navegação. A utilização destes sensores é frequente uma vez que estes conseguem estimar a posição e orientação de um corpo a três dimensões independentemente de sinais externos. Além disso, os sensores inerciais são fortemente imunes a interferências eletromagnéticas, interferências meteorológicas e são ainda de fácil implementação [5]. O uso destes sensores tornou-se mais procurado com o advento dos MEMS (*Micro-Electro-Mechanical Systems*).

De forma sumária, MEMS é um termo aplicado para descrever uma nova área de estudo, os microssistemas eletro-mecânicos. Estes sistemas são fabricados por processos de micro-fabrico não convencionais uma vez que, ao contrário da microeletrónica, onde os circuitos são estruturas sólidas e compactas, os MEMS possuem cavidades, canais, membranas, entre outros, de forma a imitar as partes mecânicas dos sensores [26]. Estes sensores, devido às suas características de baixo consumo energético e pequenas dimensões, encontram aplicações em sistemas de navegação para aeronaves, automóveis, mísseis, telemóveis entre outras [5].

Esta fusão de sensores pode ser encontrada em situações que outras fontes de informação não podem ser consideradas devido ao ambiente onde o veículo circula. Um exemplo de um sistema deste género foi utilizado no veículo Odin Figura 7.a, que participou no DARPA (*Defense Advanced Research Projects Agency*) *Urban Challenge* pela equipa *VitorTango's*. O DARPA *Urban Challenge* é uma competição de automóveis autónomos criada nos EUA de forma a incentivar o desenvolvimento tecnológico [27]. Outro exemplo pode ser visto na Figura 7.b. O ATRV-Jr produzido por *Real World Interface*. Este foi adaptado e utilizado para testar a eficiência desta fusão sensorial [28].



Figura 7 - (a) Odin, veículo usado pela equipa VitorTango's no DARPA Urban Challenge (adaptado de [27]) e (b) ATRV-Jr desenvolvido por Panzieri *et al.* [28].

### 2.3. Utilização conjunta de GPS, INS e Odometria

Os sistemas de localização compostos por **GPS, INS e Odometria** utilizam estes sensores para determinar a localização do veículo. Os sensores de eleição para registar a velocidade das rodas são os *encoders* incrementais. Alguns exemplos deste tipo de sistemas de localização podem ser encontrados em [29], [30], [31], [32] e [33].

Um *encoder* (Figura 8) é um transdutor capaz de converter um movimento linear ou rotativo num sinal tipicamente digital. Tem aplicações quando se requer um posicionamento preciso, por exemplo, das cabeças de impressão de uma impressora. Relativamente ao seu funcionamento, os *encoders* podem ser absolutos ou incrementais e relativamente à sua construção podem ser magnéticos ou óticos [5].



Figura 8 - *Encoder* Absoluto EMS22A [34].

Os *encoders* incrementais de dois canais geram dois sinais elétricos (ondas quadradas) defasados  $90^\circ$  entre si, consoante o movimento do seu eixo. Através de *software* é possível registar as transições que ocorrem durante determinado tempo. Assim é possível determinar a velocidade de rotação de um eixo. Uma vez que são incrementais, em caso de perda de energia, toda a informação recolhida previamente é perdida, necessitando, geralmente, de uma nova calibração [5][35].

Os *encoders* absolutos, em vez de gerarem duas ondas quadradas possuem um código binário associado a cada ângulo (dentro da resolução do sensor). Assim, cada posição remete para um valor único e, em caso de perda de energia, essa informação não é perdida, estando disponível no momento em que a energia é retomada [5][35].

Nestes sistemas de localização, a odometria e os INS determinam a localização entre atualizações de posições fornecidas pelo GPS. A Figura 9 ilustra o funcionamento dos sistemas de

localização deste género e é possível visualizar que, entre leituras do GPS, a localização é feita à custa da última leitura de GPS e das leituras dos INS e da odometria.

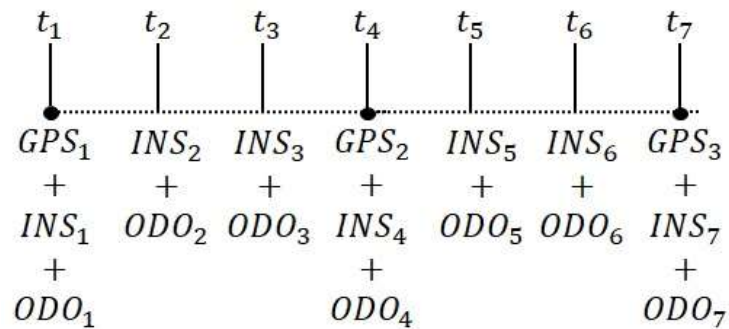


Figura 9 – Diagrama temporal do sistema de localização baseado na utilização conjunta do GPS, INS e odometria.

Um sistema de localização deste género foi utilizado no veículo da Figura 10, adaptado pela Universidade de Tsinghua [33] ou no veículo da *Google* (Figura 3)



Figura 10 - Veículo adaptado pela Universidade de Tsinghua [33].

Ao longo deste capítulo foram analisados alguns sistemas de localização baseados em diferentes sensores. Quanto maior o número de sensores maior o número de dados recolhidos e, por sua vez, a redundância destes. Em contrapartida, tal implica um maior custo e complexidade associado ao sistema global. No capítulo seguinte, serão apresentados os principais fundamentos teóricos nos quais esta dissertação se baseia.





## 3. Fundamentos Teóricos

Neste capítulo são descritos os fundamentos teóricos em que esta dissertação se baseia, tais como métodos de localização, representação de orientação, filtros matemáticos para realizar a fusão sensorial e os protocolos de baixo nível.

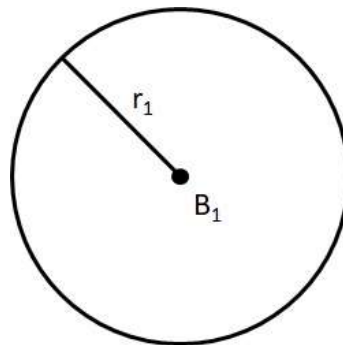
### 3.1. Métodos de localização

#### 3.1.1. Localização baseada em trilateração com balizas

Como mencionado anteriormente, as balizas podem ser definidas como dispositivos que permitem determinar de forma absoluta e contínua a posição (em alguns casos, também a orientação) de um veículo. Esta localização é possível enquanto se mantiver uma comunicação em linha de vista (*line of sight*) com uma ou mais balizas.

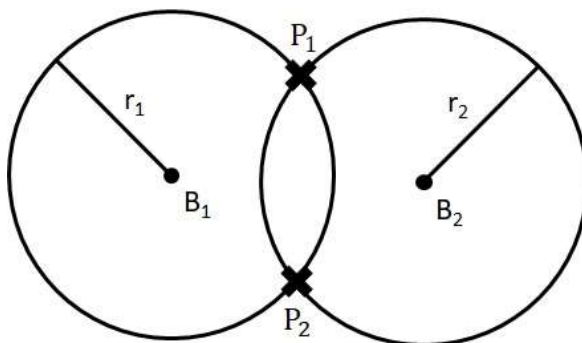
A Figura 11, a Figura 12 e Figura 13 ilustram, respetivamente, a autolocalização do veículo usando a trilateração com uma, duas ou três balizas. Na primeira situação, é possível verificar que, usando uma única baliza, obtém-se a distância do veículo a essa mesma baliza ( $r_1$ ). Esta medição simples resulta numa infinidade de pontos (circunferência de raio  $r_1$ ) em que o veículo possa estar, e é de todo impossível calcular a sua posição e orientação. Na segunda situação, com duas balizas, obtém-se a distância do veículo a cada uma das balizas –  $r_1$  e  $r_2$  – resultando em duas circunferências com os respetivos raios. Fazendo a intersecção destas duas circunferências obtém-se duas possíveis posições onde o veículo possa estar ( $P_1$  e  $P_2$ ). Por fim, na última situação, verifica-se que, usando três balizas é possível obter a distância do veículo a cada uma das balizas –  $r_1$ ,  $r_2$  e  $r_3$  – resultando em três circunferências com os respetivos raios. Fazendo uma vez mais a intersecção das três circunferências obtém-se a única posição possível para o veículo ( $P_1$ ). É importante ainda notar que se forem usadas três (ou mais) balizas colineares a ambiguidade resultante da situação dois é mantida.

O GPS é um exemplo de um sistema que utiliza este método para realizar a localização. Contudo, uma vez que existe sempre um erro associado às medições, a intersecção das circunferências gera uma área, e não um ponto, correspondente à localização do recetor [4].



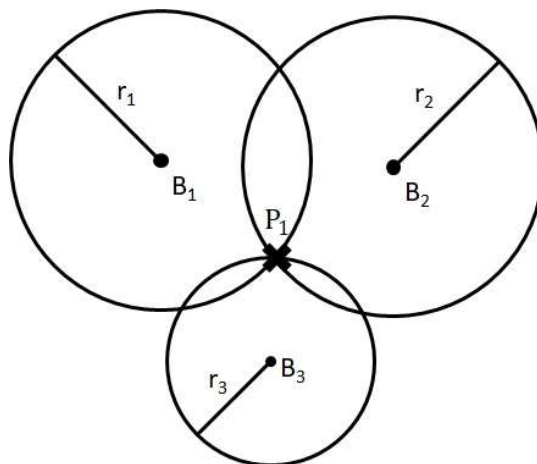
Legenda:  
 $r_1$  – Distância do veículo à baliza 1  
 $B_1$  – Baliza 1

Figura 11 - Localização usando uma baliza.



Legenda:  
 $r_{1,2}$  – Distância do veículo à baliza 1 e 2  
 $B_{1,2}$  – Baliza 1 e baliza 2  
 $P_{1,2}$  – Duas possíveis localizações do veículo

Figura 12 - Localização usando duas balizas.



Legenda:  
 $r_{1,2,3}$  – Distância do veículo à baliza 1, 2 e 3  
 $B_{1,2,3}$  – Baliza 1, baliza 2 e baliza 3  
 $P_1$  – Localização do veículo

Figura 13 - Localização usando três balizas.

### 3.1.2. Utilização de sensores inerciais

Os INS, compostos por sensores inerciais, apenas permitem a localização relativa e só podem ser usados por curtos períodos de tempo devido aos erros inerentes ao seu funcionamento. No caso do acelerómetro, de acordo com [4][5]:

- Existe uma fraca relação entre sinal e ruído em baixas acelerações. Esta situação pode levar a confundir-se o próprio ruído presente nas leituras do sensor com uma curva descrita a baixa velocidade;
- Apresenta alta sensibilidade de leitura resultando em erro de deriva (*drift error*). Qualquer alteração do plano horizontal do sensor (veículo passar por cima de um obstáculo) faz com que o sensor detete essa alteração. Barshan e Durrant-Whyte, citados em [5], adicionaram um sensor de inclinação para corrigir esta situação. Contudo, os erros de deriva obtidos pelo acelerómetro (1 a 8 cm/s) são inaceitáveis para a maior parte dos veículos autónomos;
- Os sinais têm de ser integrados duas vezes para se obter a posição do veículo. Assiste-se, assim, a um crescimento sem limites dos erros de posição do veículo com o aumento da distância percorrida (Gu *et. al.*, 2002 e Venet *et. al.*, 2002 citados por J. Esteves em [4]).

De acordo com [4][36], o giroscópio tende a ser uma melhor solução que o acelerómetro. Contudo, os sensores de maior qualidade são mais dispendiosos. Por outro lado, os mais económicos apresentam enormes erros de deriva, isto é, mesmo que o veículo se encontre parado o sensor indica que se encontra em movimento. Como o sinal tem de ser integrado para se obter a posição do veículo, verifica-se, também, um crescimento sem limites dos erros de orientação do veículo com o passar do tempo.

### 3.1.3. Odometria

A odometria baseia-se na determinação da distância percorrida pelo veículo, da orientação e velocidade do mesmo, recorrendo à medição da rotação das rodas e é uma das formas de localização mais usada em veículos. Essa medição é feita, normalmente, usando *encoders* colocados no eixo de rotação das rodas. Este método é recomendado quando a informação

recolhida pelos *encoders* representa uma fonte de informação fiável. Os erros inerentes à odometria encontram-se divididos em duas categorias: erros sistemáticos e erros aleatórios. J. Borenstein e L. Feng em [37] enumeram alguns dos (i) erros sistemáticos e (ii) aleatórios como:

(i) Erros sistemáticos:

- Diferença entre o diâmetro das rodas;
- Rodas mal alinhadas;
- Ponto de contacto entre a roda e o solo;
- Limitações físicas do sensor, como a resolução e a frequência de amostras.

(ii) Erros aleatórios:

- Superfícies irregulares;
- Navegação sobre obstáculos inesperados;
- Derrapagem das rodas.

Mesmo contendo estes dois tipos de erros, J. Borenstein *et. al.* [5], defendem que a odometria é usada porque:

- Os dados da odometria podem ser fundidos com outros dados provenientes, por exemplo, de um recetor GPS ou de um giroscópio, para se obter uma posição mais exata;
- A odometria pode ser usada entre marcos. Uma odometria mais precisa implica que não é necessária a existência de tantos marcos durante o trajeto do veículo;
- Em alguns casos, como a utilização conjunta do GPS e da odometria, esta última pode ser a única fonte de informação disponível quando ocorrem bloqueios de sinal ou condições ambientais adversas.

Os *encoders* são de fácil implementação, relativamente económicos e fortemente imunes a interferências eletromagnéticas e interferências meteorológicas [5]. Um exemplo prático do uso deste sistema é o caso do automóvel da *Google* (Figura 3).

### 3.2. Representação de Orientação

#### 3.2.1. Matriz de Rotação

A **matriz de rotação** ou **DCM** (*Direct Cosine Matrix*) define a rotação de um referencial relativamente a outro. São geralmente definidos dois referenciais (Figura 14) denominados referencial do corpo,  $O_{xyz}$ , e referencial global,  $O_{XYZ}$ , sendo que ambos os referenciais possuem a mesma origem fixa  $O$ . Sejam  $(i, j, k)$  os vetores unitários com a mesma direção dos eixos  $(x, y, z)$  e  $(I, J, K)$  os vetores unitários com a mesma direção dos eixos  $(X, Y, Z)$  [38].

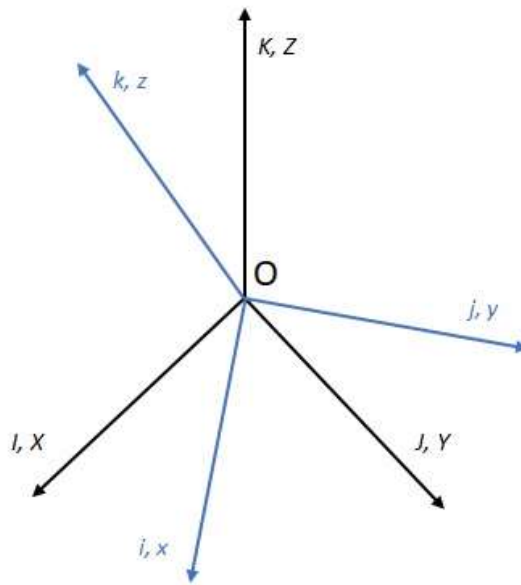


Figura 14 - Representação do referencial global e do referencial de corpo.

Sejam os vetores  $v_1 = (x_1, y_1, z_1)$  e  $v_2 = (x_2, y_2, z_2)$ . Seja  $\langle v_1 \cdot v_2 \rangle$ , o seu produto interno,  $|v|$  a norma de um vetor e  $\theta$  o ângulo entre os dois vetores, tais que:

$$|v| = \sqrt{x^2 + y^2 + z^2} \quad (1)$$

$$\langle v_1 \cdot v_2 \rangle = x_1x_2 + y_1y_2 + z_1z_2 = |v_1||v_2|\cos(\theta) \quad (2)$$

A DCM é então definida por:

$$DCM = \begin{bmatrix} \langle I.i \rangle & \langle I.j \rangle & \langle I.k \rangle \\ \langle J.i \rangle & \langle J.j \rangle & \langle J.k \rangle \\ \langle K.i \rangle & \langle K.j \rangle & \langle K.k \rangle \end{bmatrix} = \begin{bmatrix} \cos(I,i) & \cos(I,j) & \cos(I,k) \\ \cos(J,i) & \cos(J,j) & \cos(J,k) \\ \cos(K,i) & \cos(K,j) & \cos(K,k) \end{bmatrix} \quad (3)$$

Sendo que as combinações de  $I, i, J, j, K$  e  $k$  representam um ângulo. Esse ângulo é o produto entre os respectivos dois vetores.

### 3.2.2. Ângulos de Euler

A representação através dos **ângulos de Euler** define que um referencial  $\{B\}$  pode ser obtido através da rotação sequencial de um referencial  $\{A\}$  em torno dos seus três eixos  $(x, y, z) = (roll, pitch, yaw) = (\alpha, \beta, \gamma)$  relativamente ao novo referencial obtido [38]. Na Figura 15 é possível visualizar um referencial  $\{B\}$  obtido através da rotação sequencial de um referencial  $\{A\}$  onde  $R$  representa a rotação de um referencial para o outro,  $A$  o referencial inicial,  $A'$  e  $A''$  dois referencias intermediários e  $B$  o referencial final.

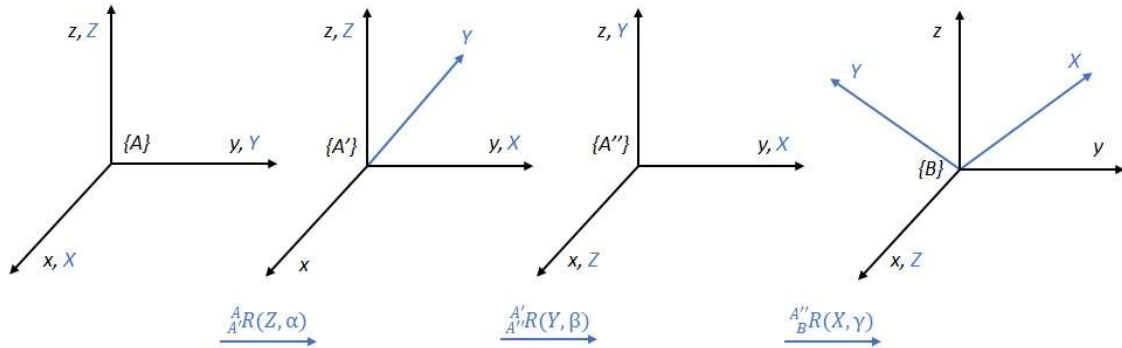


Figura 15 - Referencial  $\{B\}$  obtido através da rotação sequencial dos três eixos do referencial  $\{A\}$ .

Assim, a matriz de rotação é obtida através do produto matricial das três rotações elementares,  $c \equiv \cos$  e  $s \equiv \sin$ .

$$\begin{aligned} {}^A R_{XYZ}(\gamma, \beta, \alpha) &= R(Z, \alpha) * R(Y, \beta) * R(X, \gamma) = \\ &= \begin{bmatrix} c\alpha & -s\alpha & 0 \\ s\alpha & c\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} c\beta & 0 & s\beta \\ 0 & 1 & 0 \\ -s\beta & 0 & c\beta \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\gamma & -s\gamma \\ 0 & s\gamma & c\gamma \end{bmatrix} = \end{aligned}$$

$$= \begin{bmatrix} c\alpha * c\beta & c\alpha * s\beta * s\gamma - s\alpha * c\gamma & c\alpha * s\beta * c\gamma + s\alpha * s\gamma \\ s\alpha * c\beta & s\alpha * s\beta * s\gamma + c\alpha * c\gamma & s\alpha * s\beta * c\gamma - c\alpha * s\gamma \\ -s\beta & c\beta * s\gamma & c\beta * c\gamma \end{bmatrix} \quad (4)$$

Para facilitar a representação do cálculo de  $(\alpha, \beta, \gamma)$  assumamos que:

$${}^A_B R_{XYZ}(\gamma, \beta, \alpha) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (5)$$

$$\beta = \tan^{-1}\left(\frac{-r_{31}}{\sqrt{r_{11}^2 + r_{21}^2}}\right) \quad (6)$$

$$\alpha = \tan^{-1}\left(\frac{r_{21}/c\beta}{r_{11}/c\beta}\right) \quad (7)$$

$$\gamma = \tan^{-1}\left(\frac{r_{32}/c\beta}{r_{33}/c\beta}\right) \quad (8)$$

É importante notar que  $\beta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$  e que, uma vez que a representação se baseia na rotação sequencial de um referencial em torno dos três eixos, certas combinações de rotações podem levar ao alinhamento de dois eixos causando a perda de um eixo de rotação. Este fenómeno é conhecido como *Gimbal Lock* [38]. A Figura 16 ilustra este fenómeno onde é possível ver o alinhamento dos três *gimbals* e a perda do eixo *roll*.

<sup>1</sup> Um *gimbal* é um suporte articulado que permite a rotação de um objeto sobre um eixo.



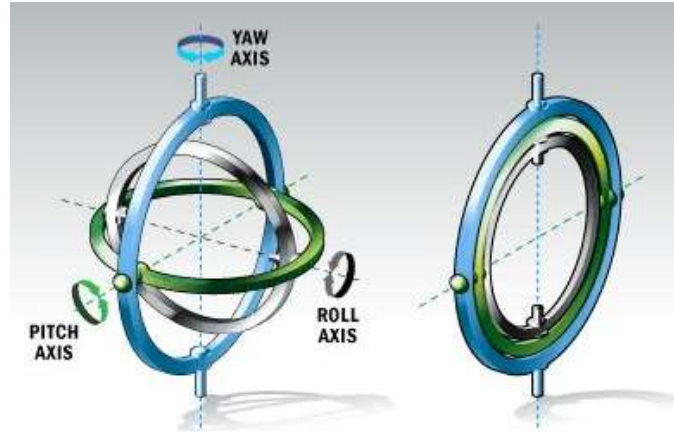


Figura 16 - Fenômeno *Gimbal Lock* (Adaptado de [39])

### 3.2.3. Vetor de Rotação

Como alternativa ao uso de rotações sequenciais dos três eixos do referencial, é possível obter um referencial  $\{B\}$  através da rotação de um referencial  $\{A\}$  a partir de um ângulo  $\theta$  em torno de um eixo  $k = [k_x \ k_y \ k_z]$  (Figura 17). [38].

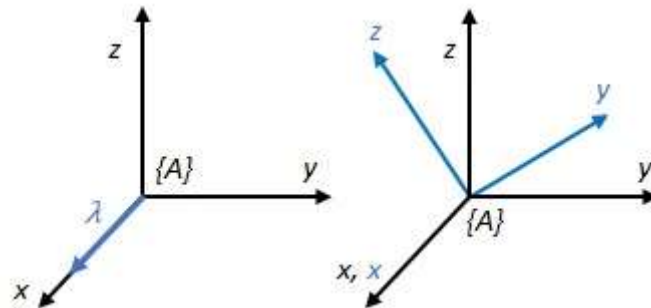


Figura 17 - Rotação de um referencial  $\{A\}$  através de um vetor de rotação  $\lambda$  para obter um referencial  $\{B\}$ .

O vetor de rotação  $\lambda$  é definido como:

$${}^A_B \lambda = \theta * \begin{bmatrix} k_x \\ k_y \\ k_z \end{bmatrix} \quad (9)$$

Sendo  $\theta \in [0, \pi]$  e  $k$  é o vetor de norma unitária que define o eixo de rotação.

### 3.2.4. Quaterniões

Introduzidos por Hamilton em 1843, os **quaterniões** são um sistema de números representados por:

$$a + bi + cj + dk = 0 \quad (10)$$

Sendo que,  $(a, b, c, d)$  são os números reais e  $(i, j, k)$  são símbolos que podem ser interpretados como vetores unitários que apontam ao longo dos três eixos espaciais. Em particular, a unidade quaternião forma uma unidade esfera em  $\mathbb{R}^4$  e

$$b^2 + c^2 + d^2 = 1 \quad (11)$$

Uma vez que a especificação de uma rotação diretamente através de quaterniões é pouco intuitiva, o procedimento normal é guardar os valores em ângulos de *Euler* e posteriormente converter em quaterniões. Desta forma, e respeitando a equação 11, a matriz de rotação,  $R(q)$ , pode ser definida como [40][41][42]:

$$q_1 \equiv \vec{x} * \sin \frac{\theta}{2} \quad (12)$$

$$q_2 \equiv \vec{y} * \sin \frac{\theta}{2} \quad (13)$$

$$q_3 \equiv \vec{z} * \sin \frac{\theta}{2} \quad (14)$$

$$q_4 \equiv \cos \frac{\theta}{2} \quad (15)$$

$$R(q) = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2 * (q_1 * q_2 + q_3 * q_4) & 2 * (q_1 * q_3 - q_2 * q_4) \\ 2 * (q_1 * q_2 - q_3 * q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2 * (q_2 * q_3 + q_1 * q_4) \\ 2 * (q_1 * q_3 + q_2 * q_4) & 2 * (q_2 * q_3 - q_1 * q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \quad (16)$$

Sendo que,  $\vec{x}$ ,  $\vec{y}$  e  $\vec{z}$  são os vetores unitários com a mesma orientação que os eixos do referencial  $O_{XYZ}$ .

Desta forma, a representação de uma rotação de um referencial  $\{A\}$  é feita através de um eixo de rotação  $\vec{q} = (q_1, q_2, q_3)$  e a partir de um ângulo  $\theta$  (Figura 18).

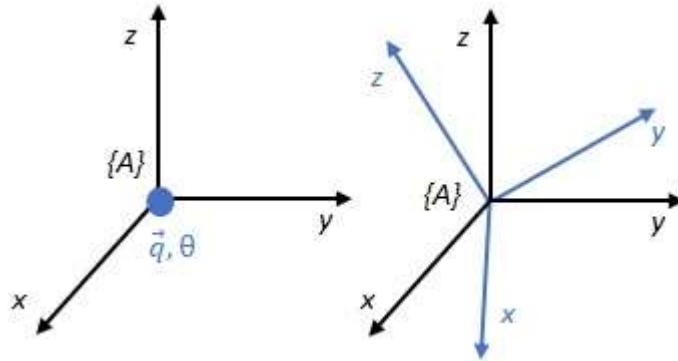


Figura 18 - Rotação de um referencial  $\{A\}$  através de um eixo de rotação  $\vec{q}$  e um ângulo  $\theta$  obtendo o referencial  $\{B\}$ .

Em suma, a DCM envolve nove elementos para ser definida. Por outro lado, os ângulos de *Euler* são capazes de representar uma rotação apenas com três. No entanto, a transformação através dos ângulos de *Euler* não é direta tendo de recorrer à DCM e determinadas combinações de rotações podem levar ao fenómeno do *Gimbal Lock*. Relativamente aos quaterniões, estes não estão sujeitos a singularidades e, uma vez que não utilizam funções trigonométricas, são mais fáceis de calcular. Contudo, o seu significado não é intuitivo [40][42].

### 3.3. Filtros Matemáticos

Neste capítulo encontram-se descritos alguns dos filtros matemáticos mais usados neste tipo de aplicações. Estes algoritmos são, na maior parte, algoritmos probabilísticos por serem tipicamente soluções mais robustas face às limitações sensoriais, como o ruído nas leituras, a resolução dos sensores ou a dinâmica do ambiente, entre outras. Uma vez que representam a informação através de distribuições probabilísticas apresentam melhores resultados em ambientes não estruturados e complexos. No entanto, são geralmente mais complexos e computacionalmente mais exigentes que os algoritmos não probabilísticos.

### 3.3.1. Filtro Complementar

O **filtro complementar** é um termo geralmente aplicado em ambiente académico para fazer referência a qualquer filtro/ algoritmo matemático que “*tenha como objetivo a fusão de dados redundantes ou similares de diferentes sensores de forma a alcançar a melhor estimativa possível de uma determinada variável*” sendo o único exemplo aqui apresentado de um filtro não probabilístico [43].

É um filtro comumente utilizado devido à resposta rápida, simplicidade de implementação, facilidade de parametrização e baixo custo computacional [44]. Um exemplo comum da utilização deste filtro é na fusão dos dados provenientes de um giroscópio e de um acelerómetro de forma a obter a orientação de um objeto. Neste caso, os dados provenientes do acelerómetro são filtrados através de um filtro passa baixo enquanto que os dados provenientes do giroscópio são integrados e filtrados por um filtro passa alto. Por fim, faz-se um somatório das duas parcelas, resultando na melhor estimativa do ângulo. Para o exemplo prático referido o filtro apresenta a seguinte formulação matemática [43][45][46][47],

$$var = data_1 * [1 - G(s)] + data_2 * G(s) \quad (17)$$

Sendo que,  $var$  é a variável de interesse,  $data_1$  e  $data_2$  são os dados provenientes do giroscópio e do acelerómetro, respetivamente,  $[1 - G(s)]$  um filtro passa alto e  $G(s)$  um filtro passa baixo.

### 3.3.2. Filtro de Kalman

O **filtro de Kalman**, desenvolvido por Rudolf Kalman em 1960, é um estimador linear recursivo capaz de estimar os valores ao longo do tempo do estado de um processo de forma a minimizar a média dos erros quadráticos de estimação. Esta estimativa é feita levando em consideração o modelo e medições do sistema, descrições estatísticas do ruído do sistema, das medições e qualquer informação adicional sobre as condições iniciais das variáveis de interesse. É importante notar que os modelos usados têm de ser lineares e que as descrições estatísticas têm de ser distribuições gaussianas, caso contrário o filtro não funcionará corretamente. Assim, o filtro de Kalman é utilizado em problemas onde é necessário fundir diferentes dados provenientes

de diferentes sensores – sistemas MISO (*Multi-Input/Single-Output*) ou sistemas MIMO (*Multi-Input/Multi-Output*) – ou em problemas onde a medição direta dos estados de interesse é impossível ou existe uma mistura de ambos [48][49].

O princípio base do filtro de Kalman consiste em utilizar a informação sobre a dinâmica do sistema, isto é, como o sistema se comporta ao longo do tempo, de forma a prever qual o próximo estado. Um exemplo muito simples permite perceber o que é feito: “*se sei onde estive e quão rápido me desloco então consigo prever onde estou.*” [49] A Figura 19 traduz o ciclo de Kalman e as suas funções.

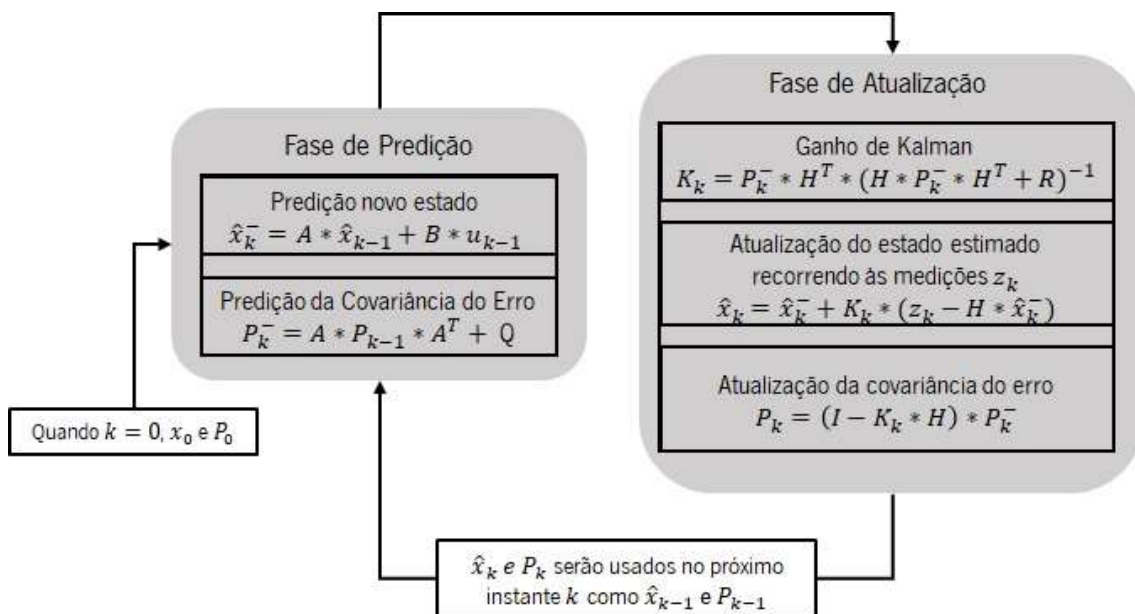


Figura 19 – Ciclo do filtro de Kalman.

O filtro de Kalman inicia-se definindo um estado inicial  $x_0$ , que é uma matriz  $n \times 1$  e  $n$  é o número de variáveis de estado, e uma matriz inicial de covariância dos erros de estimação  $P_0$ , que é uma matriz  $n \times n$ . São ambas utilizadas apenas na primeira iteração do filtro. Cada ciclo de cálculo do filtro, inicia-se com a fase de predição onde se realiza a previsão do estado com base na estimativa do último estado conhecido, a dinâmica do sistema (matriz  $A_{n \times n}$ ) e um sinal de controlo  $u$ , caso exista. A matriz  $B_{n \times l}$ , onde  $l$  é o número de sinais de controlo, relaciona a entrada de controlo  $u$  com o estado  $x$ . Por fim, é realizada a previsão da matriz de covariância dos erros levando em consideração a dinâmica do sistema, a última matriz de covariância

calculada e a covariância das perturbações atuando no sistema ou o erro do modelo do sistema (matriz  $Q_{n \times n}$ ).

A segunda fase do filtro, denominada de fase de atualização, consiste em atualizar a estimativa do estado com base na previsão previamente efetuada e com medições realizadas. Para tal, é necessário calcular o ganho de Kalman (matriz  $K_{n \times m}$  que minimiza o erro da covariância *a posteriori*) e que leva em consideração a matriz de covariância estimada  $P_k^-$ , a matriz de observação  $H_{m \times n}$  (que relaciona a medição  $z_k$  com o estado  $x$ ) e a matriz de covariância dos erros nas medições  $R$  (cuja dimensão depende do número de saídas de interesse, tendo no máximo  $n$  saídas). O ganho de Kalman, durante a atualização do estado, é responsável por fazer uma soma pesada entre o valor estimado na fase de predição e o valor obtido através das medições. Por fim, a matriz de covariância é atualizada com base no ganho de Kalman, a matriz de observação e a matriz de covariância prevista. É importante notar que o estado atualizado  $\hat{x}_k$  e a matriz de covariância  $P_k$  são usados na iteração seguinte como  $\hat{x}_{k-1}$  e  $P_{k-1}$ , sendo por isso um filtro recursivo.[48][49]. A Figura 20 ilustra o funcionamento do filtro de forma conceitual. Aqui, é possível visualizar uma previsão (azul), com média  $z_1$  e variância  $\sigma_{z1}$ , uma medição (vermelho), com média  $z_2$  e variância  $\sigma_{z2}$ , e a estimativa do estado final (preto), com média  $z_3$  e variância  $\sigma_{z3}$  (esta variância é sempre menor que a variância da previsão e da medição).

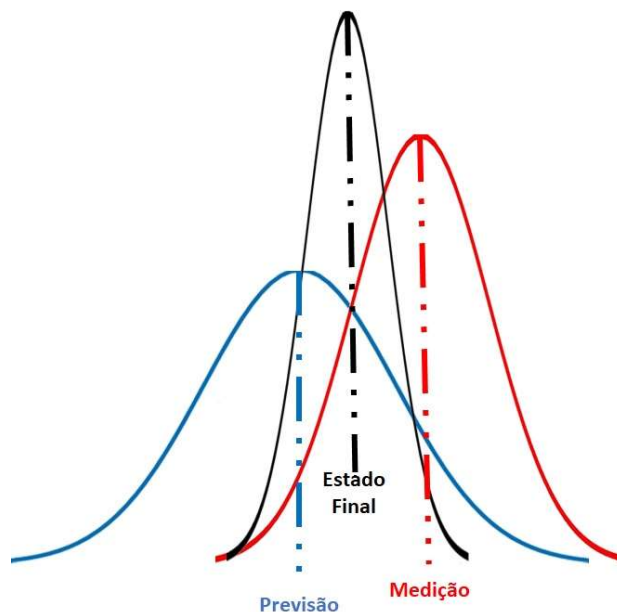


Figura 20 - Visão conceitual do filtro de Kalman.

### 3.3.3. Filtro de Kalman *Extended*

O **filtro de Kalman *Extended* (FKE)**, como o nome sugere, é uma extensão ao filtro de *Kalman* capaz de lidar com sistemas não lineares. Essa não linearidade é tratada através de uma abordagem analítica que recorre à expansão da série de Taylor. A transformação do filtro de *Kalman* para o FKE encontra-se na Figura 21 sendo que,  $f$  é a função não linear da dinâmica do sistema,  $h$  a função não linear da observação,  $w_k$  a variável aleatória que representa o ruído do processo e  $v_k$  a variável aleatória que representa o ruído das medições. Para lidar com a não linearidade, o FKE recorre à linearização local, isto é, as matrizes Jacobianas de  $A = \frac{\partial A}{\partial x}$  e  $H = \frac{\partial H}{\partial x}$  são calculadas de forma a linearizar as funções em cada iteração do filtro. Este cálculo pode tornar o filtro computacionalmente pesado caso o cálculo das matrizes seja feito através de métodos numéricos e não analíticos [50][51][52].

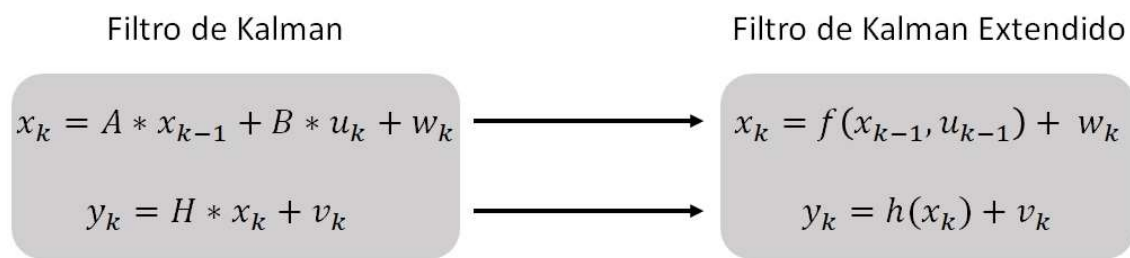


Figura 21 - Transformação das fórmulas matemáticas do filtro de Kalman para o filtro de Kalman Extendido.

### 3.3.4. Filtro de Kalman *Unscented*

O **filtro de Kalman *Unscented* (FKU)** é uma outra variante do filtro de Kalman e, como o FKE, é capaz de lidar com sistemas não lineares, contudo, recorrendo a uma abordagem estatística e não analítica. Como técnica central para lidar com a não linearidade o filtro utiliza a Transformação *Unscented*.

A transformação *Unscented* é uma função matemática usada para estimar o resultado de uma transformação não linear aplicada a uma distribuição probabilística. Para tal, utiliza um número pequeno de pontos/amostras, também denominados como *sigma points*, selecionados com base em condições *a priori*. Estas amostras são propagadas pela função não linear e a média e covariância da função podem ser determinadas. Desta forma, a distribuição gaussiana, que até ao momento era desconhecida, pode ser predita. Por fim, esta distribuição é utilizada para calcular

o novo estado estimado. É importante notar que para se realizar a recolha das amostras alguns parâmetros devem ser definidos. Estes parâmetros são conhecidos como parâmetros de dimensionamento da Transformada *Unscented* [53][54][55]:

- $\alpha$  – Determina a amplitude das amostras, isto é, a distância a que estão da média  $\mu$ . Pode assumir valores de  $10^{-4} < \alpha < 1$ ;
- $\beta$  – Usado para incluir informação sobre a distribuição prévia. Para distribuições gaussianas é utilizado  $\beta = 2$ ;
- $k$  – Geralmente colocado a 0. Tal como o  $\alpha$  influencia a amplitude das amostras;
- $\lambda = \alpha^2 * (L + K) - L$ , sendo que  $L$  é o tamanho do vetor de estados.

O algoritmo do FKU inicia-se com a definição destes parâmetros para o cálculo do peso correspondente a cada amostra. Assim, seja  $W_m$  os pesos correspondentes à média e  $W_c$  os pesos correspondentes à covariância:

$$W_m[0] = \frac{\lambda}{(L+\lambda)} \quad (18)$$

$$W_c[0] = W_m[0] + (1 - \alpha^2 + \beta) \quad (19)$$

$$W_m[i] = W_c[i] = \frac{1}{2*(L+\lambda)}, \quad i = 1, \dots, 2L \quad (20)$$

Em seguida, e como no filtro de Kalman, definem-se as matrizes  $Q, R, x_0$  e  $P_0$ , que correspondem à matriz dos erros do estado, matriz das incertezas das medições, estado inicial e a matriz de covariância inicial, respetivamente.

Após a inicialização segue-se a fase de predição que tem como objetivo estimar o novo estado e a matriz de covariância levando em consideração o conhecimento do estado anterior. Assim  $(2 * L + 1)$  amostras são obtidas através de:

$$\chi_{m_{k-1}} = [\hat{x}_{k-1} \quad \hat{x}_{k-1} + \sqrt{L + \lambda} * \sqrt{P_{k-1}} \quad \hat{x}_{k-1} - \sqrt{L + \lambda} * \sqrt{P_{k-1}}] \quad (21)$$



Sendo que,  $\chi_m$  é uma matriz  $L \times (2L + 1)$  de amostras, cada coluna da matriz representa uma amostra e  $P_k$  a matriz de covariância. É importante notar que existem diversos métodos para calcular a raiz quadrada de uma matriz, mas, por motivos de desempenho e eficiência computacional o método de *Cholesky* é recomendado para este tipo de aplicações. A raiz quadrada da matriz através desse método é,

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix} * \begin{bmatrix} l_{11} & l_{21} & l_{31} \\ 0 & l_{22} & l_{32} \\ 0 & 0 & l_{33} \end{bmatrix} \equiv LL^T \quad (22)$$

$$LL^T = \begin{bmatrix} l_{11}^2 & l_{21}l_{11} & l_{31}l_{11} \\ l_{21}l_{11} & l_{11}^2 + l_{22}^2 & l_{31}l_{21} + l_{32}l_{22} \\ l_{31}l_{11} & l_{31}l_{21} + l_{32}l_{22} & l_{11}^2 + l_{22}^2 + l_{33}^2 \end{bmatrix} \quad (23)$$

$$\begin{cases} l_{11} = \sqrt{a_{11}} \\ l_{22} = \sqrt{a_{22} - l_{21}^2} \\ l_{33} = \sqrt{a_{33} - (l_{31}^2 + l_{32}^2)} \\ l_{21} = \frac{a_{21}}{l_{11}} \\ l_{31} = \frac{a_{31}}{l_{11}} \\ l_{32} = \frac{(a_{32} - l_{31}l_{21})}{l_{22}} \end{cases} \quad (24)$$

Após gerar todas as amostras é feita a propagação destas pela função não linear, sendo que  $\chi_{m_k}$  corresponde às amostras propagadas,  $u_k$  ao sinal de controlo:

$$\chi_{m_k}^{(i)} = f(\chi_{m_{k-1}}^{(i)}, u_k, Q) \quad (25)$$

O estado e covariância estimados são dados pelas expressões 26 e 27, respetivamente:

$$\hat{x}_k^- = \sum_{i=0}^{2L} \chi_{m_k}^{(i)} * W_m[i] \quad (26)$$

$$P_k^- = Q + \sum_{i=0}^{2L} W_c[i] * (\chi_m - \hat{x}_k^-) * (\chi_m - \hat{x}_k^-)^T \quad (27)$$

Segue-se a fase de observação que consiste na propagação das amostras através das medições realizadas, isto é, de acordo com as medições da evolução do estado ao longo do tempo,

$$\psi_m^{(i)} = h(\chi_{m_k}^{(i)}, u_k, R), \quad i = 0, 1, \dots, 2L \quad (28)$$

$$\hat{y}_k = \sum_{i=0}^{2L} W_m[i] * \psi_m^{(i)} \quad (29)$$

Sendo que,  $\psi_m$  é a matriz de saída de amostras propagadas através da função de observação e  $\hat{y}_k$  a média da saída predita. São calculadas as matrizes de covariância da saída predita  $P_{yy}$  e a matriz de covariância entre o estado e a saída  $P_{xy}$ , respetivamente,

$$P_k^{yy} = R + \sum_{i=0}^{2L} W_c[i] * (\psi_m^{(i)} - \hat{y}_k)(\psi_m^{(i)} - \hat{y}_k)^T \quad (30)$$

$$P_k^{xy} = \sum_{i=0}^{2L} W_c[i] * (\chi_{m_k}^{(i)} - \hat{x}_k^-)(\psi_m^{(i)} - \hat{y}_k)^T \quad (31)$$

Por fim, é realizado a fase de atualização do estado levando em consideração as medições realizadas. Assim, o ganho de Kalman  $K_k$  é calculado através das duas matrizes de covariância previamente calculadas.

$$K_k = \frac{P_k^{xy}}{P_k^{yy}} \quad (32)$$

Como em todos os casos, o ciclo do filtro acaba com a atualização do estado  $\hat{x}_k$  e da matriz de covariância  $P_k$ ,

$$\hat{x}_k = \hat{x}_{k-1} + K_k * (z_k - \hat{y}_k) \quad (33)$$

$$P_k = P_k^- - K_k * P_k^{yy} * K_k^T \quad (34)$$

Sendo que,  $z_k$  corresponde às medições realizadas.

Em forma de conclusão, a Tabela 1 resume algumas características dos filtros apresentados. Pela análise da mesma é perceptível que quando se abandonam os modelos lineares pelos modelos não lineares a complexidade e exigência computacional dos filtros aumenta. É também visível que o filtro de Kalman *Unscented* é mais exigente computacionalmente, mas é aquele capaz de lidar com modelos totalmente não lineares.

Tabela 1 - Comparação entre os diferentes filtros matemáticos (adaptado de [56]).

Filtro	Modelo	Distribuição	Custo Computacional
Complementar	Linear	—	Muito baixo
Kalman	Linear	Gaussiana	Baixo
Kalman Extendido	Localmente linear	Gaussiana	Baixo se as matrizes Jacobianas forem calculadas analiticamente; Médio se forem calculadas numericamente.
Kalman Unscented	Não linear	Gaussiana	Médio/Alto

### 3.4. Protocolos de comunicação de baixo nível

Os protocolos de comunicação de baixo nível são todos aqueles protocolos utilizados para realizar a comunicação entre os diferentes sensores, módulos de comunicação e o microcontrolador. Estes são regras que permitem a duas ou mais entidades trocarem informação. O protocolo pode definir regras de sintaxe, semântica, sincronização de comunicação e métodos de recuperação de erros. Estes protocolos podem ser implementados por *hardware*, *software* ou uma combinação de ambos [57][58]. Os protocolos explorados e utilizados no projeto são UART, I2C e SPI.

#### 3.4.1. UART

Um ***Universal Asynchronous Receiver/Transmitter (UART)*** é um protocolo de comunicação responsável por transformar uma comunicação paralela<sup>2</sup> numa comunicação série<sup>3</sup> ou vice-versa (Figura 22). É usado, por exemplo, para fazer a comunicação entre o CPU (*Central*

<sup>2</sup> Comunicação paralela é o processo de enviar vários bits em simultâneo através de um canal de comunicação. O número de bits possíveis de enviar em simultâneo depende do número de canais de dados existentes [60][86].

<sup>3</sup> Comunicação série é o processo de enviar dados bit a bit, de forma sequencial, através de um canal de comunicação [62][76].

*Processing Unit*) do microcontrolador e os sensores/atuadores ligados a este. Desta forma, a comunicação entre o microcontrolador e os diferentes dispositivos é realizada através de uma comunicação série assíncrona<sup>4</sup> utilizando duas linhas de dados (RX e TX). Internamente, entre o CPU e a UART, é realizada uma comunicação paralela. A UART foi desenvolvida para comunicar entre dois dispositivos de cada vez sendo uma comunicação *full-duplex*<sup>6</sup> [59][60].

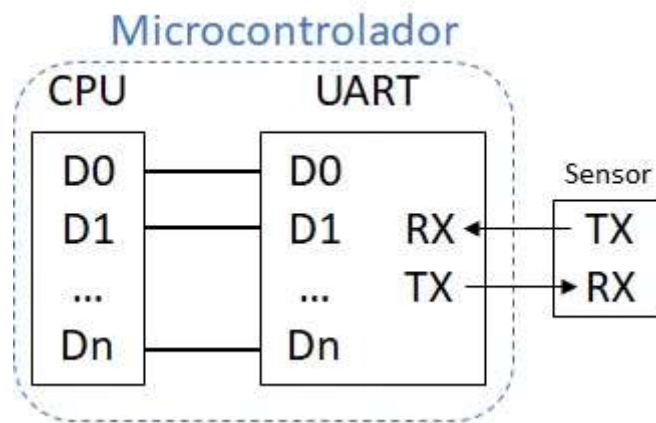


Figura 22 – Esquema de uma comunicação UART.

### 3.4.2. I2C

A **Inter-Integrated Circuit (I2C)** ou **Two-Wire Interface (TWI)** é uma interface de comunicação série síncrona<sup>6</sup> *half-duplex*<sup>7</sup>. Foi desenvolvida pela *Phillips* para permitir comunicações entre um dispositivo denominado *Master*, que controla o fluxo de informação, e um ou vários dispositivos denominados *Slave*, que apenas comunicam quando o *Master* o permite. Para a comunicação são apenas necessárias duas linhas: a linha SDA, que é a linha de dados, e a linha SCL, que é linha do relógio. O I2C foi desenvolvido para comunicações de curtas distâncias (normalmente entre periféricos no mesmo PCI) e permite taxas de comunicação geralmente entre

<sup>4</sup> O envio de dados não se encontra sincronizado com nenhum sinal de relógio [58].

<sup>5</sup> Refere-se à capacidade de o envio e receção de dados poder ser realizado em simultâneo entre duas entidades [58].

<sup>6</sup> O envio de dados encontra-se sincronizado com um sinal de relógio [58].

<sup>7</sup> Refere-se à capacidade do envio e receção de dados poder ser realizado nos dois sentidos, contudo, nunca em simultâneo [58].

os 100 kHz e os 400 kHz. A Figura 23 ilustra uma conexão entre um *Master* e vários *Slave* [61][62].

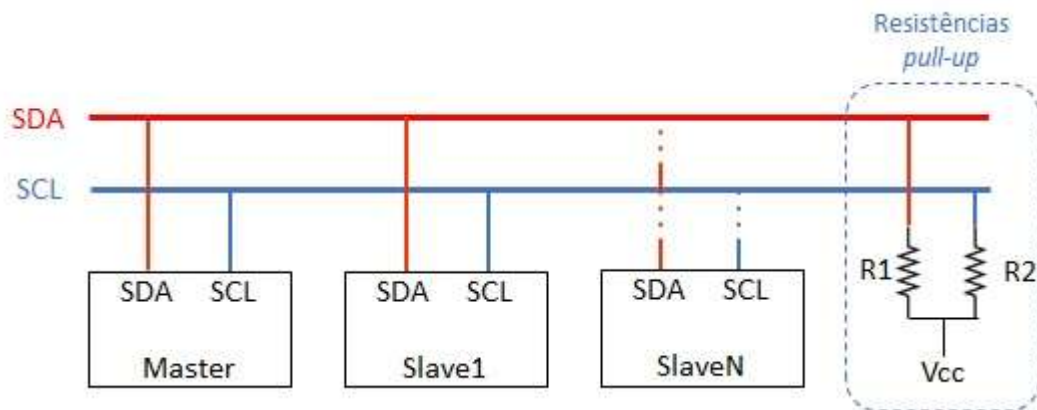


Figura 23 - Esquema de um barramento I2C.

### 3.4.3. SPI

A **Serial Peripheral Interface (SPI)** é uma interface de comunicação série síncrona *full duplex*. Similar ao I2C, é uma interface *Master-Slave*, contudo, necessita no mínimo de quatro ligações físicas sendo que três delas estão ligadas simultaneamente a todos os dispositivos (Figura 24) [61][58]:

- **Synchronous Clock (SCK)** – linha de relógio para sincronizar todos os dispositivos. Esta linha é controlada pelo *Master*;
- **Master Out/Slave In (MOSI)** – linha usada pelo *Master* para transferir dados para os *Slaves*;
- **Master In/Slave Out (MISO)** – linha usada pelos *Slaves* para transferir dados para o *Master*.

As outras linhas que podem existir são relativas ao **Slave-Select (SS)** e são responsáveis por selecionar qual dos *Slaves* é que vai comunicar com o *Master*. A lógica presente nestas linhas é denominada *active-LOW* porque, quando o *Master* está a comunicar com um *Slave* (ou vice-versa) a linha SS respetiva a esse *Slave* é colocada a *LOW* e todas as outras linhas SS existentes são colocadas a *HIGH* [61][58].

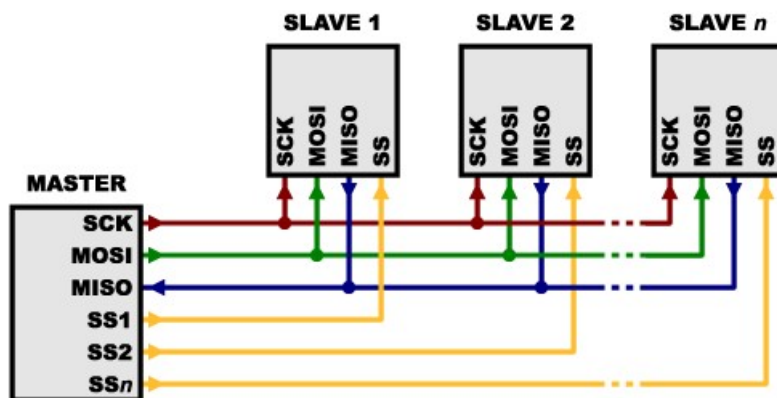


Figura 24 - Esquema de uma conexão SPI [58].



## 4. Arquitetura do sistema de localização

Neste capítulo descreve-se a arquitetura do protótipo do sistema de autolocalização. O sistema tem como principal objetivo realizar uma fusão sensorial de forma a obter a melhor estimativa possível sobre a localização do veículo. Um dos requisitos foi o desenvolvimento de uma solução de baixo custo. Desta forma, o trabalho pode ser dividido em três tarefas principais:

1. **Desenvolvimento da Unidade de Recolha de Dados**
  - a. Desenvolvimento do *hardware*;
  - b. Desenvolvimento do *firmware*.
2. **Desenvolvimento da Unidade de Processamento de Dados**
  - a. Implementação dos diferentes filtros matemáticos.
3. **Desenvolvimento da interface gráfica.**

Na Figura 25 encontra-se um diagrama que ilustra a arquitetura do sistema desenvolvido, evidenciando os principais módulos.

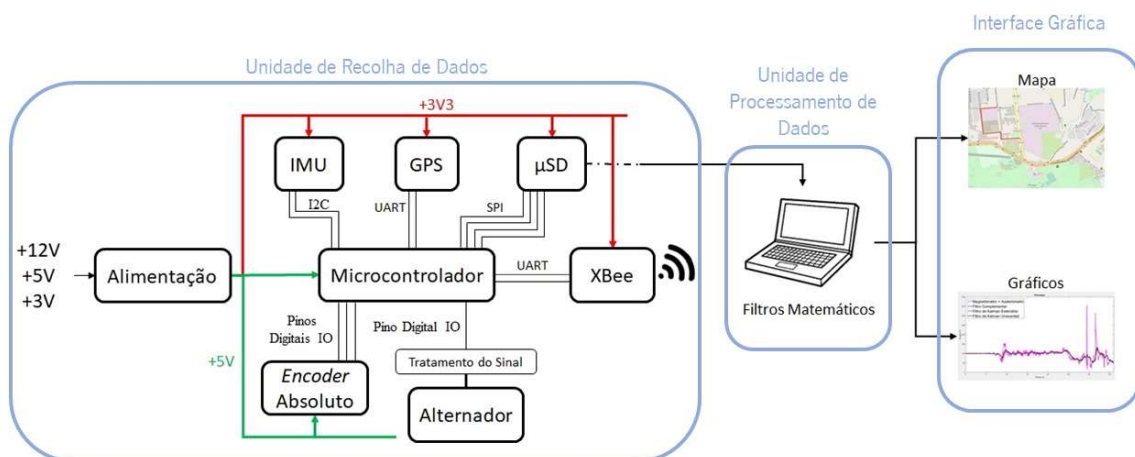


Figura 25 - Arquitetura geral do sistema implementado, evidenciando os diferentes níveis de tensão, os sensores utilizados e protocolos de comunicação.



#### 4.1. Unidade de Recolha de Dados

A **Unidade de Recolha de Dados (URD)**, como o nome indica, é responsável por realizar toda a recolha de dados necessária para determinar a posição e a orientação do veículo. Foi necessário selecionar sensores capazes de fornecer informação que auxilie essa determinação. Para tal, e levando em consideração o requisito de um sistema de baixo custo, selecionou-se:

- Um recetor GPS para fornecer a posição absoluta em exteriores;
- Um *encoder* absoluto para determinar o ângulo da direção;
- Um *encoder* incremental para registar a velocidade de uma das rodas;
- Um acelerómetro para registar as acelerações lineares do veículo;
- Um giroscópio para medir a velocidade de rotação do veículo;
- Um magnetómetro, por fornecer a orientação relativamente ao norte magnético da Terra.

Quanto à comunicação de dados, esta é assegurada através de duas formas distintas: através de uma comunicação com fios e através de uma comunicação sem fios. A utilização destas duas formas distintas tem como objetivo explorar as vantagens deste tipo de comunicações para a aplicação em causa. Por fim, relativamente à alimentação do sistema, este conta com três níveis de tensão distintos: +3 V, +5 V (com conector microUSB) e +12 V. Estes níveis de tensão foram escolhidos porque +12 V é o nível de tensão *standard* das baterias dos veículos, facilitando a integração do sistema; +5 V e +3 V pela facilidade de obtenção através de baterias para telemóveis ou pilhas.

#### 4.2. Unidade de Processamento de Dados

A **Unidade de Processamento de Dados (UPD)** é responsável por tratar e fundir todos os dados fornecidos pela URD recorrendo a diferentes filtros matemáticos. Nesta unidade, estão implementadas todas as equações de cinemática do veículo. É também possível comparar os diferentes resultados dos diferentes filtros e inferir qual ou quais apresentam a melhor solução. A Figura 26 ilustra a arquitetura da UPD, evidenciando que cada filtro aplicado gera um vetor de dados que é transmitido à interface gráfica.

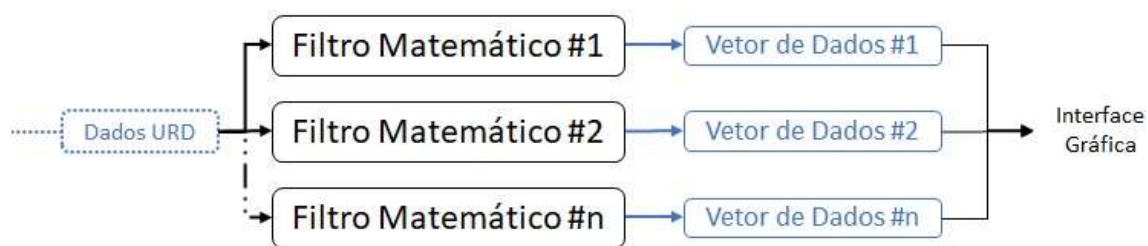


Figura 26 - Arquitetura geral da unidade de processamento de dados.

### 4.3. Interface Gráfica

A **interface gráfica** tem como intuito facilitar a visualização dos resultados. É um módulo que não interfere com o normal funcionamento dos outros dois módulos, no entanto, sem o seu correto funcionamento, é impossível a visualização dos resultados do sistema de autolocalização. A Figura 27 ilustra a arquitetura da interface gráfica encontrando-se evidenciado o seu funcionamento. A interface gráfica recebe o vetor de dados e descodifica-o de forma a realizar gráficos e desenhar as trajetórias num mapa.

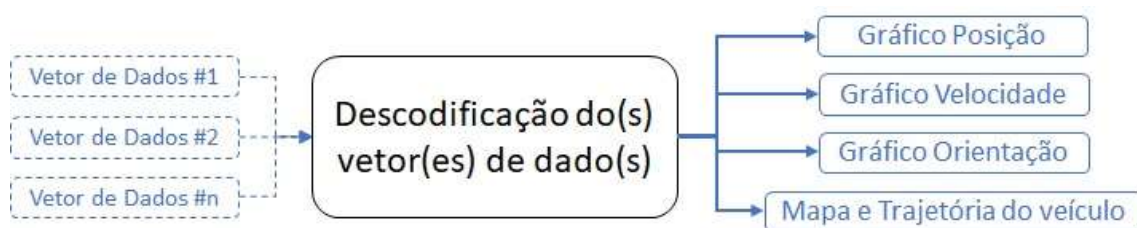


Figura 27 - Arquitetura geral da interface gráfica desenvolvida.



## 5. Protótipo do sistema de localização

Neste capítulo apresenta-se o trabalho desenvolvido a nível de *hardware*, *firmware* e *software* dos três módulos principais do sistema.

A URD desenvolvida conta com um recetor GPS Quectel L80 M39, um *encoder* absoluto Bourns EMS22A, um alternador Shimano DH-C6000 e uma IMU 9-DOF (*Degrees of Freedom*). Para a comunicação de dados a unidade conta com um cartão  $\mu SD$  e um módulo sem fios *XBee* S2C Pro. Relativamente à alimentação conta com uma bateria de +12 V com uma capacidade de 3000 mAh. No entanto, a URD encontra-se preparada para ser alimentada através de outras fontes de alimentação com valores de tensão diferentes.

A UPD foi desenvolvida em Matlab (*software* interativo de alto desempenho para cálculo numérico) e a interface gráfica em HTML (*HyperText Markup Language*), CSS (*Cascading Style Sheets*) e JavaScript.

### 5.1. Unidade de Recolha de Dados

Numa primeira fase, numa tentativa de obter uma solução mais económica, robusta e eficiente desenvolveu-se uma placa de circuito impresso (PCI) para acoplar todos os sensores e componentes da URD. Para tal, foi utilizado o *software Autodesk Eagle 8.6.0*. Na Figura 28.a) é possível visualizar parte do esquemático desenhado e na Figura 28.b) o *layout* da PCI. No apêndice A encontram-se o esquemático e *layout* completo da placa desenvolvida. Por fim, na Figura 29.a) e na Figura 29.b) apresentam-se a parte inferior e a parte superior da PCI produzida e assemblada.

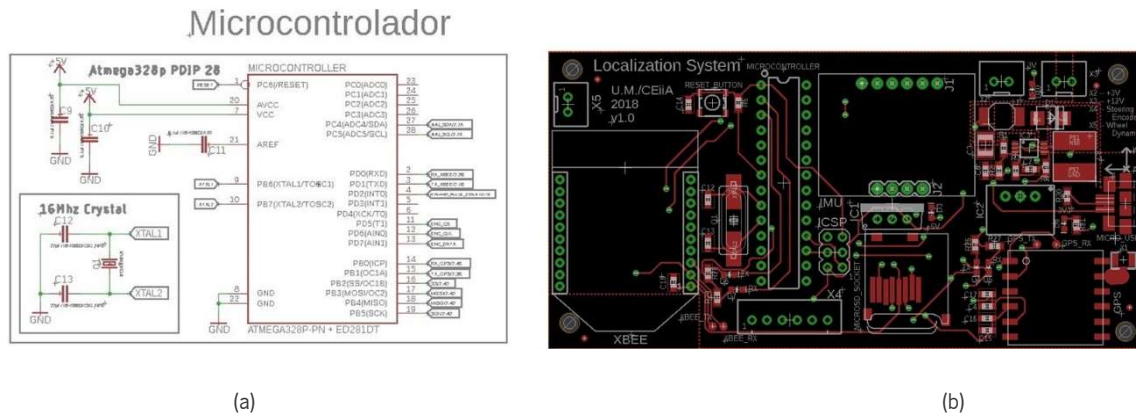


Figura 28 - a) Parte do esquemático desenhado; b) *Layout* superior da PCI desenvolvida

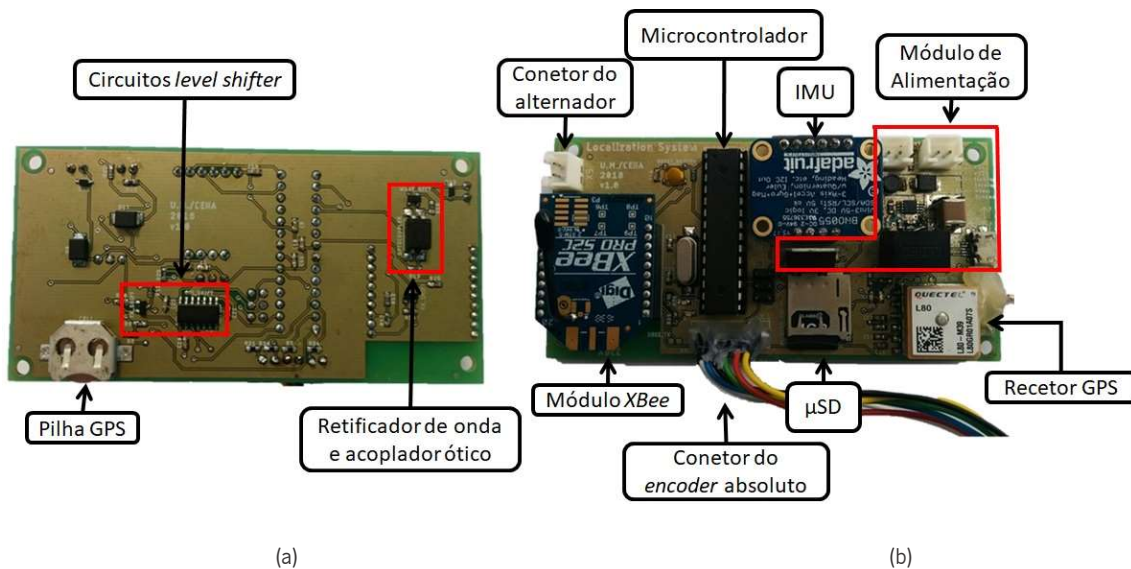


Figura 29 - a) Face inferior da PCI produzida; b) Face superior da PCI produzida.

É importante realçar que todos os componentes selecionados se encontravam ativos, isto é, ainda faziam parte do catálogo do fabricante (significa que ainda estão em produção, e, portanto, não são obsoletos), sem chumbo e cumprem a diretiva RoHs (*Restriction of Certain Hazardous Substances*). O RoHs é uma diretiva europeia com o objetivo de restringir o uso de substâncias perigosas e nocivas, presentes em muitos componentes eletrónicos, como o chumbo e o mercúrio [63].

Para realizar todo o processamento e recolha de dados foi utilizado o microcontrolador *Atmel/Atmega328p* por ir de acordo com todos os requisitos necessários (protocolos de comunicação, custo, documentação disponível, entre outros).

Relativamente ao *firmware* do microcontrolador, a Figura 30 exemplifica o seu funcionamento. É possível perceber que o sistema pode funcionar de três maneiras distintas: (i) unicamente com o cartão  $\mu SD$ ; (ii) unicamente com a *XBee*; (iii) com ambos. A URD inicia o seu funcionamento verificando a conexão com a *XBee* e/ou com o cartão  $\mu SD$ . Caso exista algum erro, esta reportará esse mesmo erro e terminará o programa. Caso contrário, e apenas para o cartão  $\mu SD$ , verifica se o nome “*d000.csv*” existe no cartão. Se esta situação se verificar, o valor “000” será incrementado até ser possível criar o ficheiro. Uma vez garantida a comunicação de dados a URD testa a comunicação com a IMU. Caso exista algum erro, a unidade reporta-lo-á e terminará o programa. Uma vez garantida a comunicação com a IMU, realiza-se a calibração da mesma (bicicleta imóvel durante uns segundos para calibrar o giroscópio e realizar 8’s no ar com a bicicleta para calibrar o magnetómetro). Enquanto a calibração não se realizar, o sistema não passará deste ponto.

Após a calibração, a URD aguarda que o GPS realize o primeiro *fix*, isto é, descubra onde se encontra na Terra (o programa da unidade aguarda neste ponto para não escrever os primeiros dados sem localizações associadas). Assim que se verifica esta condição, é ativada a interrupção externa 0 do microcontralado para registar os pulsos gerados pelo alternador. No ciclo principal, se existirem novos dados do GPS para descodificação, estes são descodificados, senão verifica se desde a última escrita já passaram 25 ms. Se isso se verificar então a URD recolhe os valores medidos pelo GPS, IMU, *encoder* e alternador e prepara-os, colocando-os numa *string*. Assim que concluída essa preparação esta é enviada pela *XBee* (através do protocolo UART) ou escrita no cartão  $\mu SD$  (através do protocolo SPI).

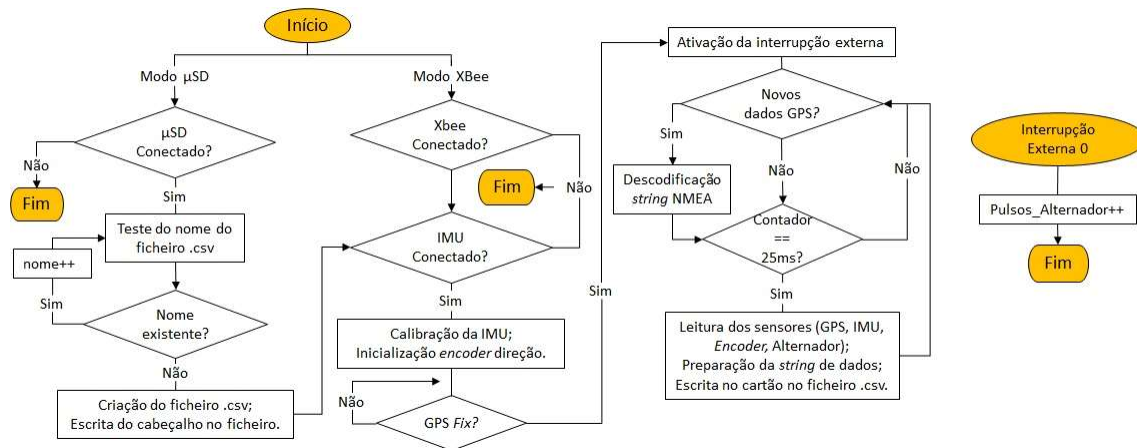


Figura 30 - Fluxograma do código desenvolvido para a unidade de recolha de dados.

Para fixar a URD na bicicleta foi necessário desenvolver uma solução simples e eficiente. Gentilmente, a equipa de Engenharia e Mobilidade de Produto do CEiiA, disponibilizou algum tempo para o desenho em CAD (*Computer-Aided Design*), e construção das peças, por impressão 3D. Na Figura 31.a) encontra-se o desenho em CAD das peças e na Figura 31.b) encontram-se evidenciadas as mesmas acopladas à bicicleta.

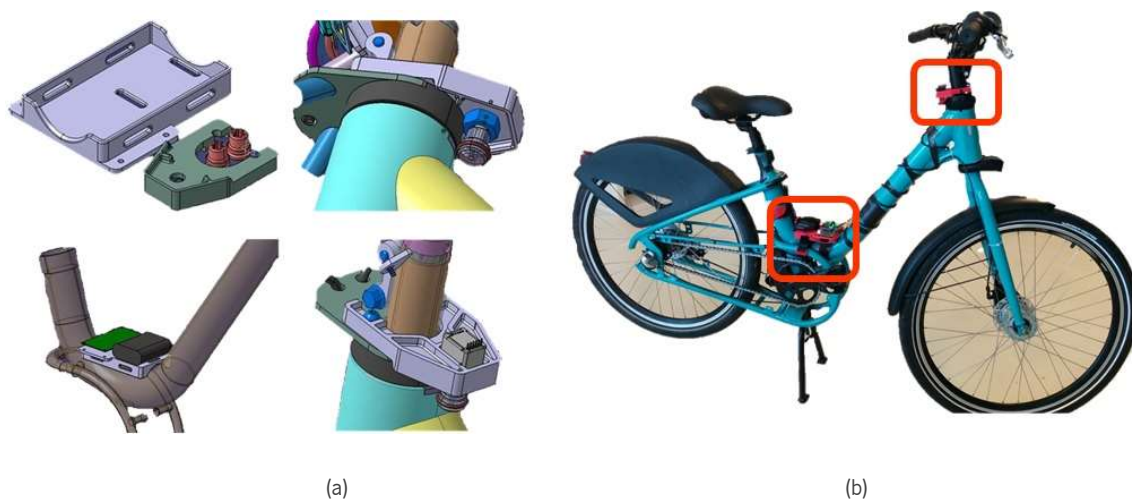


Figura 31 - a) Desenho em CAD das peças para acoplar a URD à bicicleta; b) Peças acopladas à bicicleta utilizada para testes.

Os subcapítulos que se seguem têm como intuito explicar os diferentes módulos que constituem a URD e o trabalho realizado em cada um deles.

### 5.1.1. GPS

Utilizou-se um recetor GPS porque, em muitas circunstâncias, constitui uma solução adequada e de relativo baixo custo à determinação da posição absoluta em ambientes exteriores. Desde que estejam à vista quatro ou mais satélites é possível realizar o primeiro *fix* (descoberta da posição do recetor). O recetor utilizado foi o Quectel L80 M39 (Figura 32).



Figura 32 - Recetor GPS Quectel L80 M39 [64].

O recetor tem as seguintes características [64]:

- Tensão: 20mA/@+3.3V;
- Protocolo de comunicação UART: 4800bps (*bits per second*) a 19200bps (9600bps padrão)
- 66 canais de procura, 22 canais de rastreamento;
- Sensibilidade:
  - Aquisição: -148dBm;
  - Re-Aquisição: -160dBm;
  - Rastreamento: -165dBm;
- Precisão da Posição Horizontal: <2.5 m CEP<sup>8</sup> (*Circular Error Probable*);
- Taxa de atualização: 1 Hz a 10 Hz.

O recetor devolve uma *string* NMEA (*National Marine Electronics Association*) 0183 (Figura 33) que consiste numa interface de comunicação. Esta define requisitos elétricos do sinal e o formato da *string* para comunicação série entre dispositivos eletrónicos de uso naval (sonares, anemómetros, recetores GPS, entre outros) [65]. De forma a fazer a descodificação da *string* e obtenção de valores como a latitude, longitude, velocidade, entre outros, foi utilizada a biblioteca

---

<sup>8</sup> CEP – é um sistema de medição de precisão que define que 50% dos valores estão a menos de um determinado valor (neste caso, 2.5 m) [87]



TinyGPS++ da *Arduiniana* [66]. Uma vez que a única UART disponível no microcontrolador da URD é utilizada por outro módulo foi necessário emular, através de *software*, uma segunda UART. Para tal, foi utilizada a biblioteca *SoftwareSerial* do *Arduino*. A UART emulada tem um funcionamento semelhante à nativa até uma *baudrate*<sup>9</sup> de 19200 bps.

```
$GPRMC,16.0009.000,A,4110.3177,N,00840.6913,W,0.46,313.75,100918,,,A*75
$GPVTG,313.75,T,,M,0.46,N,0.85,K,A*31
$GPGGA,16.0009.000,4110.3177,N,00840.6913,W,1.5,2.00,129.2,M,51.3,M,,*4F
$GPGSA,A,3,01,09,17,22,23,,,,,,,,,2.21,2.00,0.94*03
$GPGSV,3,1,12,23,74,098,36,09,63,190,44,19,48,276,28,17,47,234,32*76
$GPGSV,3,2,12,03,46,060,28,06,44,311,24,33,41,190,26,22,27,074,28*70
$GPGSV,3,3,12,01,25,126,42,11,08,146,29,31,04,032,26,07,01,167,*7A
$GPGLL,4110.3177,N,00840.6913,W,16.0009.000,A,A*42
$GPTXT,01,01,02,ANTSTATUS=OK*3B
```

Figura 33 - Exemplo de uma *string* NMEA 0183 por descodificar, retirada do recetor utilizado.

Devido aos níveis de tensão de funcionamento do microcontrolador (+5 V) e do recetor GPS (+3V3) serem diferentes, para se realizar uma comunicação é necessário implementar um circuito intermediário de modo a regular os níveis de tensão. Esses circuitos são conhecidos como *level shifters* e podem ser unidireccionais (Figura 34.a) ou bidireccionais (Figura 34.b) [67]

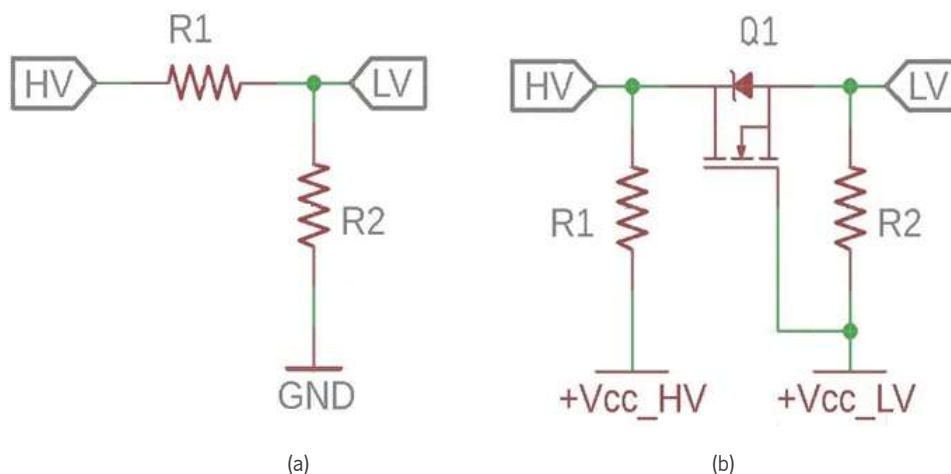


Figura 34 - a) Level shifter unidirecional; b) Level shifter bidirecional (adaptado de [67]).

<sup>9</sup> Define a velocidade de comunicação de um canal, por exemplo, 9600 bps significa que consegue enviar 9600 *bits* num segundo.

### 5.1.2. IMU

Uma **IMU** (*Inertial Measurement Unit*) é um dispositivo eletrónico, tipicamente constituída por um acelerómetro<sup>10</sup> e um giroscópio<sup>11</sup>. As mais completas podem ainda incluir um magnetómetro<sup>12</sup> e um barómetro<sup>13</sup>. O uso das IMUs em aplicações de robótica tornou-se mais comum com o aparecimento dos MEMS por serem sensores com baixo consumo energético, compactos, eficientes e económicos [27].

Para a dissertação foi utilizada a IMU *Adafruit BNO055* 9-DOF (Figura 35) por oferecer uma boa relação qualidade/preço e por possuir muita documentação disponível. A nível de *firmware* foi utilizada a biblioteca BNO055 da *Adafruit*.

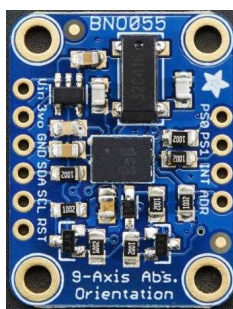


Figura 35 - *Adafruit BNO055 9-DOF* [68].

Nesta IMU pode-se encontrar um acelerómetro, um giroscópio e um magnetómetro de 3 eixos, sendo capaz de medir acelerações lineares, velocidades angulares e campos magnéticos, respetivamente. As características da IMU são [69]:

- Consumo: 12.3 mA/@5 V;
- Acelerómetro:
  - Gama de aceleração:  $\pm 2$  g/ $\pm 4$  g/ $\pm 8$  g/ $\pm 16$  g;
  - Banda do filtro passa-baixo: 1 kHz - < 8 Hz;
  - Modos de operação: Normal, Suspenso, Baixo consumo, Espera, Adormecido;
  - Modos de Trigger: Inclinação, Modo de reconhecimento lento, Detecção de altos Gs.
- Giroscópio:

<sup>10</sup> Sensor capaz de medir acelerações lineares.

<sup>11</sup> Sensor capaz de medir velocidades angulares.

<sup>12</sup> Sensor capaz de medir a intensidade de um campo magnético.

<sup>13</sup> Sensor capaz de medir a pressão atmosférica.

- Gamas programáveis:  $\pm 125$  °/s até  $\pm 2000$  °/s;
- Banda do filtro passa-baixo: 523 Hz – 12 Hz;
- Modos de operação: Iguais aos do acelerómetro;
- Modos de Trigger: Inclinação e Alta taxa.
- Magnetómetro:
  - Gama de campos magnéticos:  $\pm 1300$   $\mu\text{T}$  (x-, y-axis) e  $\pm 2500$   $\mu\text{T}$  (z-axis);
  - Resolução de  $\pm 0.3$   $\mu\text{T}$
  - Modos de operação: Baixo consumo, regular, regular melhorado, alta precisão.

### 5.1.3. Alternador

Um **alternador** é uma máquina capaz de transformar energia mecânica em energia elétrica. Algumas utilizações comuns são em centrais hidroelétricas ou em automóveis. O alternador produz a energia elétrica através do fenómeno de indução eletromagnética. Isto é, através da rotação de um rotor constituído por materiais ferromagnéticos é criado um campo magnético em rotação induzindo uma corrente elétrica nos enrolamentos do estator. Esta corrente induzida é alternada e a frequência da onda gerada está diretamente ligada ao número de pólos do alternador [70]. A Figura 36 ilustra o fenómeno descrito.

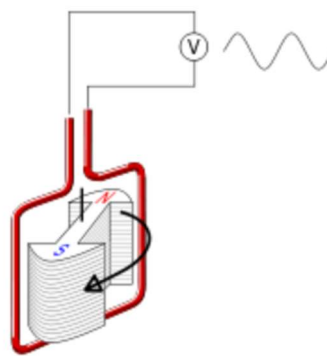


Figura 36 - Esquema de um alternador com dois pólos e a onda sinusoidal gerada por este [71].

Através da retificação da onda gerada, é possível contar o número de pulsos gerados num dado intervalo de tempo. Desta forma, é possível utilizar um alternador como um *encoder* incremental de um canal. No contexto da dissertação, utilizou-se um alternador Shimano Nexus DH-C6000 (Figura 37), com 12 pólos [72], acoplado ao eixo da roda de uma bicicleta de forma a medir a velocidade da mesma. Optou-se por um alternador em vez de um *encoder* incremental

dado que o CEiiA já possuía estes dispositivos para as suas bicicletas. Desta forma, tornou-se mais simples a instrumentação na bicicleta e reduziu-se o custo do projeto.



Figura 37 - Alternador Shimano Nexus DH-C6000 utilizado no projeto [72].

Como o sinal gerado por um alternador é uma onda sinusoidal, tendo sido necessário a implementação de algum *hardware* para transformar o sinal alternado num sinal contínuo. Assim, foi implementado um retificador de onda completa seguido de um acoplador ótico de forma a detetar os pulsos gerados com o microcontrolador. Na Figura 38 é possível visualizar o circuito implementado.

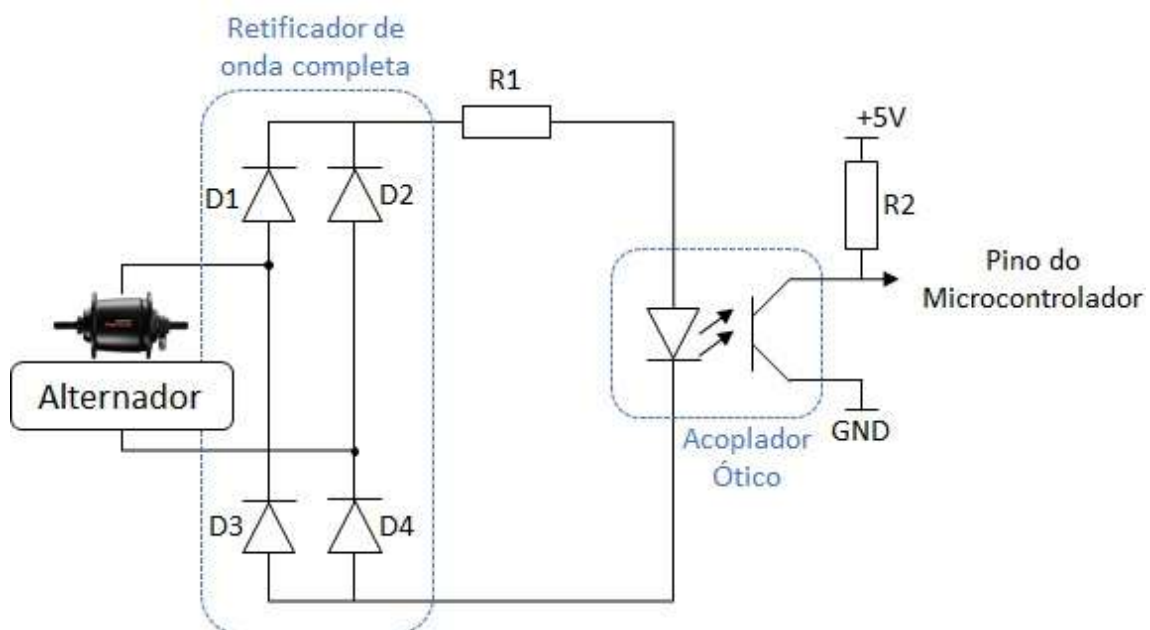


Figura 38 - Circuito eletrónico desenvolvido para transformar o sinal alternado num sinal contínuo.

A resistência R1 tem como função limitar a corrente máxima fornecida ao acoplador ótico. Sabendo que a corrente de funcionamento do componente é dada por  $1\text{ mA} \leq$

$I_{funcionamento} \leq 60 \text{ mA}$  [73], considerou-se  $R1 = 470 \Omega$ . A resistência R2 funciona como uma resistência *pull-up*: quando o transistor do acoplador ótico não conduz, o microcontrolador lê um '1' lógico; quando conduz, lê um '0' lógico.

Tendo todo o *hardware* funcional é necessário calcular o deslocamento correspondente a cada pulso gerado. Sabendo que o alternador tem 12 pólos e que foi utilizado um retificador de onda completa, as equações 35 e 36 traduzem o cálculo para obter resolução do sensor.

$$D_{pp}(cm/pulso) = \frac{PerímetroRoda}{NúmeroMáximoPulsos} \quad (35)$$

$$D_{pp}\left(\frac{cm}{pulso}\right) = \frac{208 \text{ cm}}{24} = 8.67 \text{ cm/pulso} \quad (36)$$

#### 5.1.4. Encoder Absoluto

Foi utilizado um *encoder* absoluto Bourns EMS22A (Figura 39.a) de 1024 posições [34] para registar o ângulo da direção da bicicleta. A Figura 39.b) ilustra o funcionamento implementado, isto é: se a direção curvar para esquerda o *encoder* decrementa e se curvar para a direita o *encoder* incrementa.

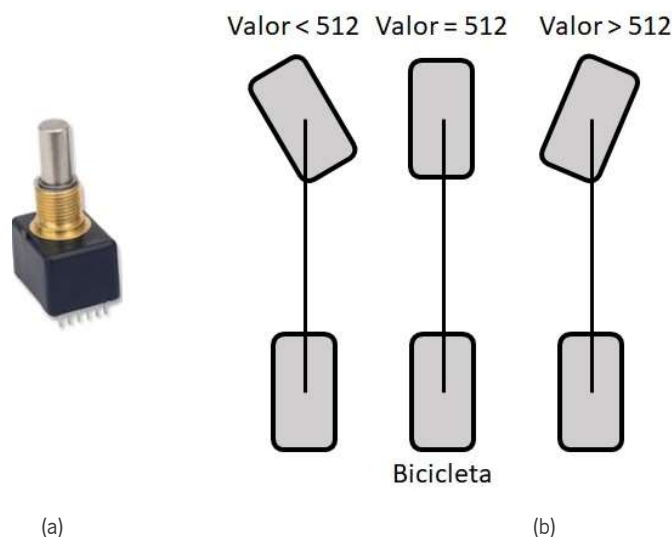


Figura 39 - a) Encoder Bourns EMS22A utilizado; b) esquema do funcionamento do encoder na unidade de recolha de dados.

Para obter a resolução do *encoder* as equações 37 e 38 foram aplicadas:

$$Resolução_{Enc} (^{\circ}/pulso) = \frac{360^{\circ}}{NúmeroMáximoPulsos} \quad (37)$$

$$Resolução_{Enc} (^{\circ}/pulso) = \frac{360}{1024} = 0.3515 ^{\circ}/pulso \quad (38)$$

A nível de *firmware*, o *encoder* em questão contém um protocolo específico para a obtenção do seu valor. Assim, é necessário emular um sinal de relógio (CLK), um sinal de *trigger* (CSn) e um sinal de dados (D0). O ciclo de leitura do *encoder* inicia-se com a realização de uma transição de *HIGH* para *LOW* na linha CSn e na linha CLK. Sempre que acontece uma transição deste género o estado da linha D0 pode ser registado (um *bit* lido de cada vez). Para obter o valor total é necessário realizar a leitura da linha D0 10 vezes. O protocolo permite ainda ler o estado e a paridade, contudo, para o contexto da dissertação, esses valores não são registados. A Figura 40 ilustra o protocolo do *encoder*.

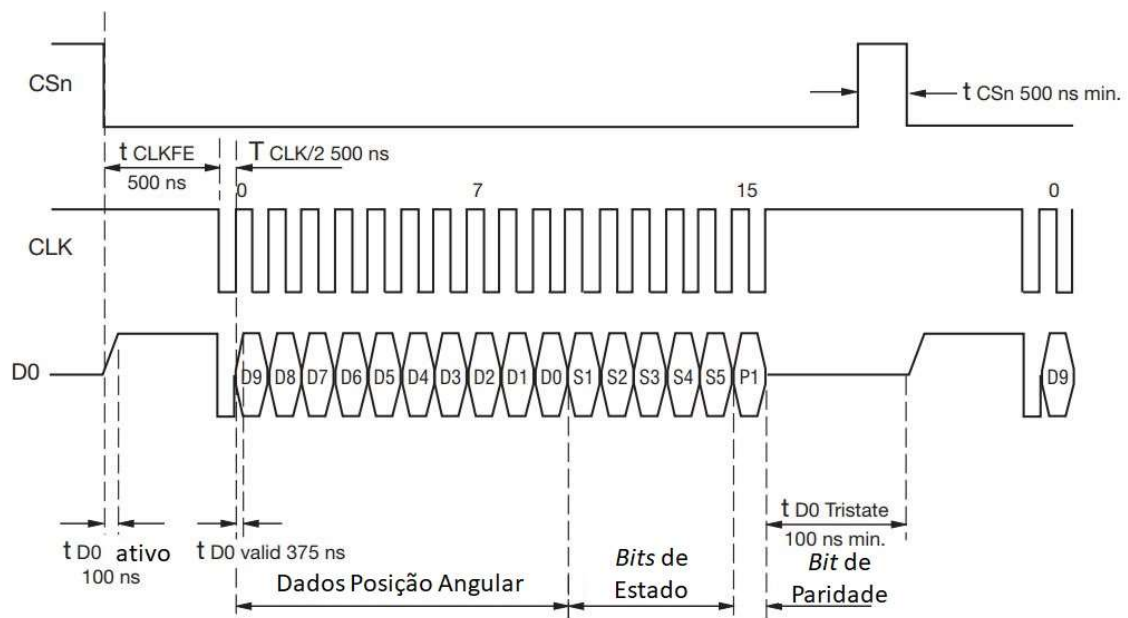


Figura 40 - Protocolo para ler o valor do encoder absoluto Bourns EMS22A [34].

### 5.1.5. Comunicação de Dados

A comunicação de dados entre a URD e a UPD encontra-se assegurada por uma comunicação com fios e sem fios. Para a comunicação com fios foi utilizado um cartão  $\mu SD$  de 8 GB *Class 10* por oferecer uma solução económica e sem interferências externas. A nível de *firmware* foi utilizada a biblioteca SD do *Arduino* e o protocolo de comunicação SPI. Uma vez que o nível de tensão de funcionamento do cartão (+3.3V) e do microcontrolador (+5 V) são diferentes foi necessário a implementação de um *level shifter*. O cartão utilizado apresenta um consumo médio de 200 mA/@3.3 V e uma velocidade máxima de escrita de 10 Mbit/seg [74]. Relativamente à comunicação sem fios foi utilizada uma *XBee S2C Pro* por permitir recolher e processar os dados em tempo real, embora se encontre limitada a nível de alcance. Foi utilizado o protocolo de comunicação UART com uma *baudrate* de 9600 bps para a comunicação entre o microcontrolador e o módulo *XBee*. O referido módulo utiliza o protocolo IEEE 802.15.4 (*Institute of Electrical and Electronics Engineers*) para comunicação entre outros módulos semelhantes e apresenta um consumo de 45 mA/@3.3 V e uma velocidade de envio de 250 Kbit/seg. Uma vez mais, foi necessário a utilização de um *level shifter* [75].

A *Digi XBee* é uma marca de módulos de comunicação sem fios da *Digi Internacional*. A empresa apresenta diversos módulos de comunicação com diferentes protocolos de comunicação, como o IEEE 802.15.4 e o IEEE 802.11. Os módulos *XBee S2C Pro* permitem construir redes de comunicação simples (ponto-a-ponto) ou redes mais complexas (redes em malha). A Figura 41 evidencia as topologias de rede possíveis de construir [75]. No projeto foi utilizada uma comunicação ponto-a-ponto.

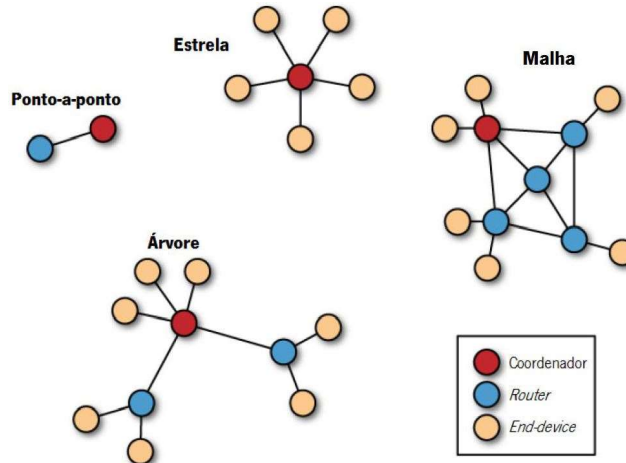


Figura 41 - Topologias de rede possíveis de construir utilizando o módulo *XBee Pro S2B* [35].

É possível ainda visualizar a função de cada dispositivo na respetiva rede. Estes podem ser [75][35]:

- **Coordenador** – Numa rede tem de existir um e um só coordenador e é responsável por criar e gerir a rede;
- **Router** – Responsável por reencaminhar dados entre módulos. Pode existir um, mais do que um ou até nenhum, dependendo da dimensão da rede;
- **End-Device** - Dispositivo mais próximo do sensor/atuador. Ao contrário do coordenador e do *router*, tem a capacidade para entrar em modo de poupança de energia, desligando momentaneamente alguns recursos.

A Figura 42 apresenta um exemplo da comunicação realizada através da *XBee* e do cartão  $\mu SD$ , respetivamente. Aqui são visíveis os diferentes parâmetros registados.



## Protótipo do sistema de localização

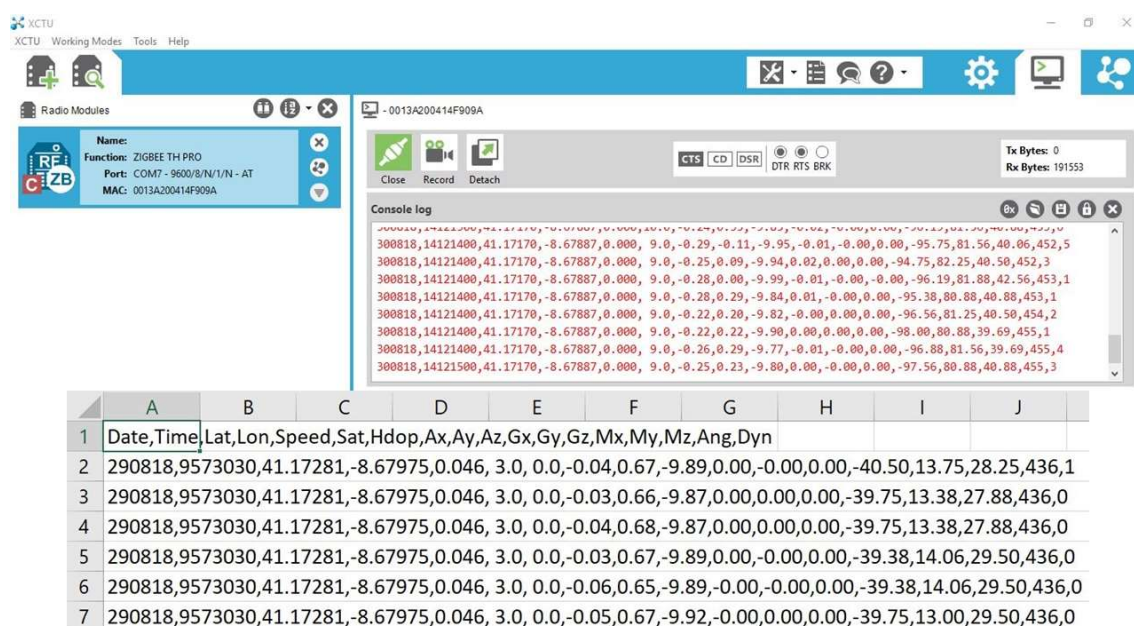


Figura 42 - Exemplo de um excerto de uma comunicação sem fios e de um ficheiro .csv.

### 5.1.6. Módulo de Alimentação

O módulo de alimentação da URD desenvolvida conta com três entradas de alimentação, +12 V, +5 V e +3 V. Estes três níveis foram considerados porque: a unidade tem como objetivo ser integrada num veículo real onde +12 V é um *standard* das baterias dos veículos; a título de desafio académico foi proposto o planeamento de formas alternativas para alimentar o sistema. Assim, +5 V pode ser obtido através do isqueiro do automóvel ou de baterias portáteis e +3 V através de pilhas convencionais.

Para regular a tensão de entrada recorreu-se a conversores de potência lineares e comutados. Os lineares, apesar de mais económicos que os comutados, possuem o seu rendimento diretamente relacionado com a diferença entre o valor de tensão de entrada e o valor de tensão de saída. Deste modo, quanto maior a diferença, menor o rendimento e vice-versa. Isto deve-se ao tipo de semicondutores que constituem este tipo de conversores. A equação 39 traduz o cálculo para determinar o rendimento de um conversor de potência linear.

$$\eta = \frac{P_{OUT}}{P_{IN}} = \frac{V_{OUT} \cdot I}{V_{IN} \cdot I} = \frac{V_{OUT}}{V_{IN}} \quad (39)$$

Sendo que,  $\eta$  é o rendimento do conversor e  $\eta \in [0,1]$ ;  $P_{OUT}$  a potência de saída do conversor;  $P_{IN}$  a potência de entrada do conversor;  $V_{OUT}$  a tensão de saída;  $V_{IN}$  a tensão de entrada;  $I$  a corrente que atravessa o conversor.

Um conversor de potência comutado apresenta rendimentos típicos na ordem dos 90% devido aos semicondutores que o constituem. Possuem, no entanto, preços consideravelmente mais elevados que os lineares. A Tabela 2, adaptada de [35], resume algumas vantagens e desvantagens dos conversores de potência.

Tabela 2 - Diferenças entre conversores eletrónicos de potência lineares e comutados (adaptado de [35]).

Tipo de Conversor	$\eta$	Vantagens	Desvantagens
Lineares	$\frac{V_{OUT}}{V_{IN}}$	Simplicidade do circuito; Baixo custo; Ausência de comutações, reduzindo o risco de interferência eletromagnética.	Baixo rendimento; Impossibilidade de $V_{OUT} > V_{IN}$ .
Comutados	Geralmente entre [75%, 95%]	Rendimento elevado; Aplicabilidade; Possibilidade de $V_{IN} \leq V_{OUT}$ (apenas em conversores com topologia do tipo <i>Boost</i> , <i>Buck-Boost</i> e SEPIC).	Risco de interferências eletromagnéticas devido a comutações de alta frequência; Complexidade do circuito; Custo.

Uma vez que os diferentes componentes eletrónicos da URD funcionam com diferentes valores de tensão, foi necessário regular a tensão de entrada para os respetivos valores de tensão. Deste modo, é necessário considerar o consumo geral da unidade para dimensionar os reguladores. A Tabela 3 resume o consumo de cada componente e o nível de tensão de funcionamento.

Tabela 3 - Consumo dos componentes utilizados na URD.

Componente	Tensão	Corrente
Microcontrolador (Atmel Atmega328p)	+5 V	12 mA
GPS (Quectel L80 M39)	+3.3 V	20 mA
IMU (Adafruit BNO055)	+5 V	12.3 mA
Encoder Absoluto (Bourns EMS22A)	+5 V	20 mA
XBee S2C Pro	+3.3 V	45 mA
Cartão µSD	+3.3 V	200 mA
$I_{total}$		309.3 mA

Considerando um conversor de potência linear obtém-se a seguinte potência dissipada:

$$P_{Dissipada} = (V_{Entrada} - V_{Saída}) * I_{total} \quad (40)$$

$$P_{Dissipada} = (12 V - 5 V) * 309.3 mA = 2.1651 W \quad (41)$$

Uma vez que a maior parte dos conversores de potência lineares não suportam elevadas potências de dissipação (dependendo do ambiente de funcionamento ou com a introdução de dissipadores a potência dissipada pode variar) e apresentam baixos rendimentos [35] foi utilizado um conversor de potência comutado (R78C) do tipo *step-down* (conversor capaz de colocar na saída um valor de tensão inferior ao que tem na sua entrada) para regular a tensão de +12 V para +3.3 V. Uma vez que a corrente consumida pelos dispositivos com uma tensão +5 V é de apenas 44.3 mA, a potência dissipada é de apenas 310 mW e, por isso, optou-se por um conversor de potência linear (LM7805) para regular a tensão de +12 V para +5 V. Por fim, e de forma a explorar academicamente os conversores de potência foi implementado um conversor de potência comutado (LM2621) do tipo *step-up* (conversor capaz de colocar na sua saída um valor de tensão superior ao que tem na sua entrada). A Figura 43 ilustra a arquitetura do sistema de alimentação implementada na URD. O diodo D1 e D2 tem como função garantir segurança ao sistema caso se liguem os três níveis de alimentação em simultâneo [35][76].

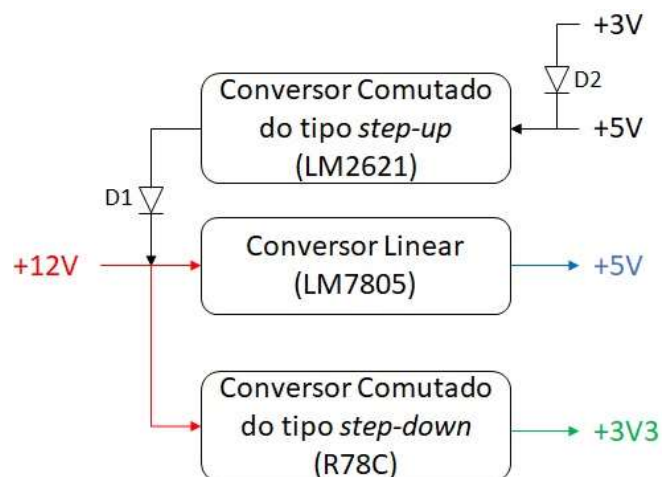


Figura 43 - Arquitetura do sistema de alimentação implementado na URD.

## 5.2. Cinemática do veículo

De forma a validar o protótipo desenvolvido foi necessária a integração deste num veículo real. Para tal, utilizou-se uma bicicleta (Figura 44), cedida pelo CEiiA, em alternativa a um automóvel. Esta decisão deveu-se ao custo, tempo requerido e dificuldade em adquirir e instrumentar um automóvel.



Figura 44 - Bicicleta utilizada para validar o protótipo do sistema de autolocalização desenvolvido.

Para além disso, para efeitos de localização, o modelo cinemático de um veículo de quatro rodas (como o automóvel tradicional) pode ser simplificado no modelo cinemático de uma

bicicleta. Esta simplificação é conhecida na literatura como *modelo da bicicleta* [77][78]. A Figura 45 ilustra os dois modelos cinemáticos, evidenciando as características comuns que tornam possível a simplificação. Assim,  $L$  é a distância entre as rodas da frente e as rodas de trás,  $\alpha$  o ângulo que as rodas da frente fazem relativamente ao veículo,  $R$  o raio de curvatura,  $\vec{v}$  a velocidade linear,  $(x, y)$  as coordenadas cartesianas do veículo e  $\theta$  a orientação do veículo.

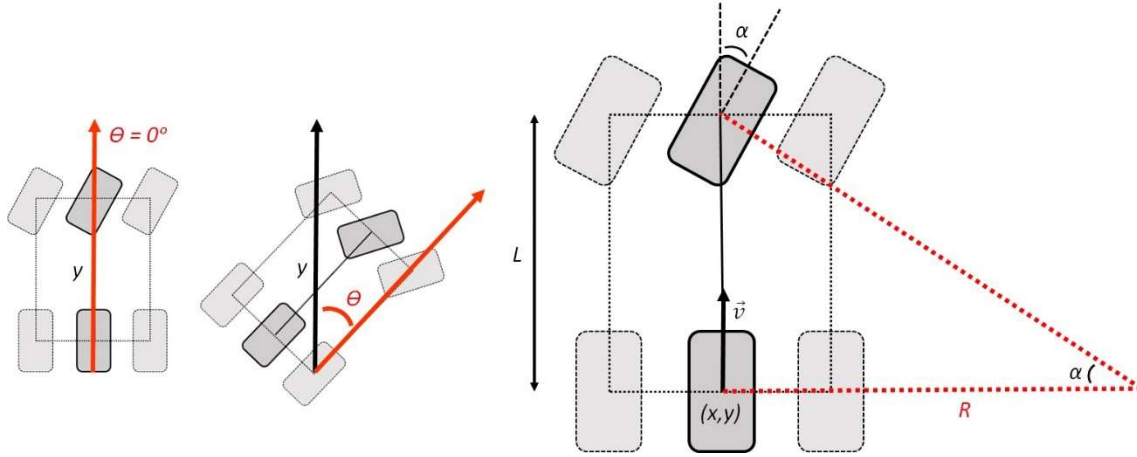


Figura 45 - Modelo cinemático do automóvel e modelo da bicicleta. As rodas do automóvel estão representadas a cinzento claro e as rodas da bicicleta a cinzento escuro.

As equações 42, 43 e 44 regem a cinemática do veículo:

$$\dot{x} = |v| * \cos(\theta) \quad (42)$$

$$\dot{y} = |v| * \sin(\theta) \quad (43)$$

$$\dot{\theta} = \frac{|v|}{L} * \tan(\alpha) \quad (44)$$

Sendo que,  $\dot{x}$ ,  $\dot{y}$  e  $\dot{\theta}$  são a velocidade em  $x$ ,  $y$  e  $\theta$ , respetivamente;  $|v|$  o módulo da velocidade linear do veículo;  $\theta$  a orientação do veículo;  $\alpha$  orientação das rodas relativamente ao veículo. É importante notar que para a bicicleta  $\alpha$ ,  $R$ ,  $\vec{v}$  e  $(x, y)$  são comuns e  $L$  representa a distância entre a roda da frente e a roda de trás.

### 5.3. Unidade de Processamento de Dados

A UPD é responsável por processar e fundir todos os dados recolhidos pela URD. Aqui é onde se implementam os filtros matemáticos e as equações relativas à cinemática do veículo, tendo sido desenvolvida utilizando o programa *Matlab*. O diagrama presente na Figura 46 resume o algoritmo desenvolvido.

Inicialmente é realizada a descodificação da *string* recebida ou do ficheiro .csv de forma a alocar cada variável com o respetivo valor. Segue-se a primeira fusão que leva em consideração os valores obtidos pelo magnetómetro e o acelerómetro. Por fim são aplicados os diferentes filtros e obtidos os vetores de valores com a latitude, longitude e orientação do veículo.

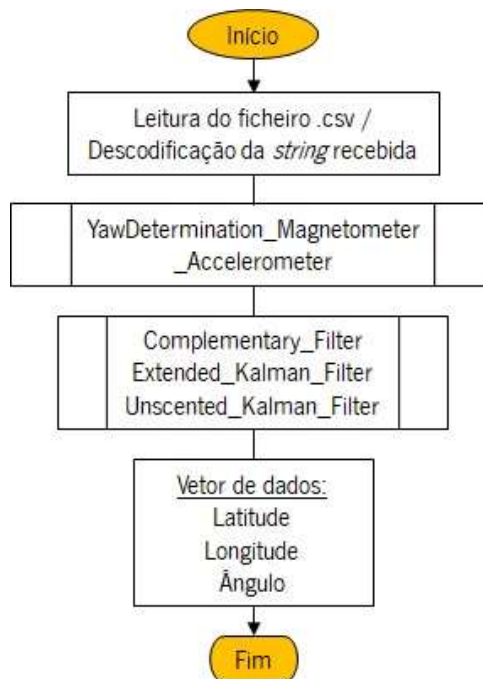


Figura 46 - Algoritmo geral da UPD.

A função *YawDetermination\_Magnetometer\_Accelerometer* tem como objetivo realizar o cálculo da orientação utilizando os valores do magnetómetro e do acelerómetro. De acordo com [79] e [80], se o magnetómetro se encontrar sobre uma superfície plana e nunca sair desta, o valor da orientação (*yaw*) pode ser calculado segundo a equação 45:

$$yaw = \tan^{-1}\left(\frac{Mag_y}{Mag_x}\right) \quad (45)$$

Sendo que,  $Mag_y$  e  $Mag_x$  correspondem aos valores medidos pelo sensor nos eixos -y e -x, respetivamente. Caso contrário, e uma vez que não é esperado que a bicicleta circule num plano perfeito é necessário aplicar uma compensação de inclinação [79][80]. O algoritmo implementado encontra-se ilustrado no diagrama da Figura 47.

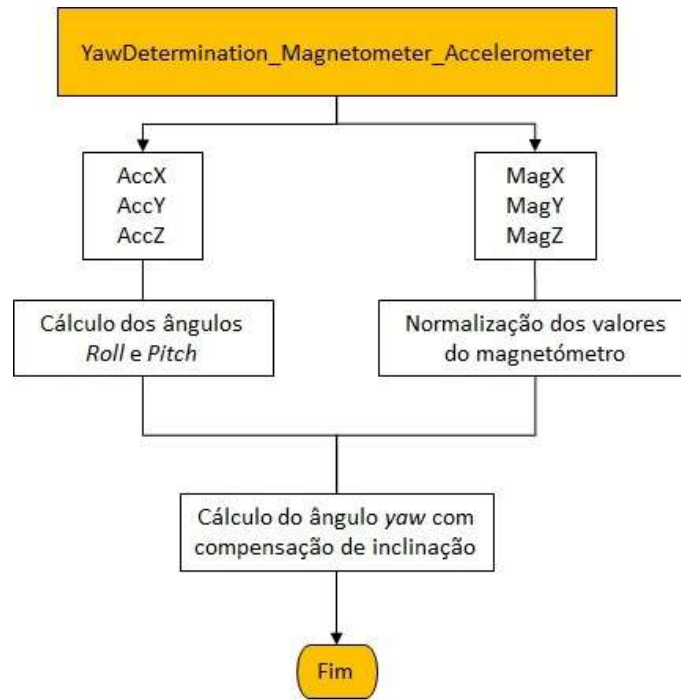


Figura 47 - Função implementada da compensação de inclinação do magnetómetro.

Inicialmente os valores do magnetómetro são normalizados, de forma a conter os valores entre -1 e 1. As equações 46, 47, 48 e 49 traduzem a normalização realizada onde,  $Mag_x$ ,  $Mag_y$  e  $Mag_z$  representam as medições do magnetómetro e  $MagN_x$ ,  $MagN_y$  e  $MagN_z$  as medições normalizadas:

$$Norm = \sqrt{Mag_x^2 + Mag_y^2 + Mag_z^2} \quad (46)$$

$$MagN_x = \frac{Mag_x}{Norm} \quad (47)$$

$$MagN_y = \frac{Mag_y}{Norm} \quad (48)$$

$$MagN_z = \frac{Mag_z}{Norm} \quad (49)$$

Relativamente aos valores do acelerómetro, estes são utilizados para determinar os ângulos de *Euler* (*Roll* e *Pitch*). As equações 50 e 51 traduzem o cálculo efetuado onde,  $Acc_x$ ,  $Acc_y$  e  $Acc_z$  representam as leituras do acelerómetro:

$$Roll = \tan^{-1} \left( \frac{Acc_y}{-Acc_z} \right) \quad (50)$$

$$Pitch = \tan^{-1} \left( \frac{Acc_x}{\sqrt{Acc_y^2 + Acc_z^2}} \right) \quad (51)$$

É importante notar que o ângulo  $pitch \in ]-90^\circ, 90^\circ[$  e ângulo  $roll \in ]-180^\circ, 180^\circ[$ . Caso seja necessário obter a orientação do veículo em 3D é recomendável a utilização de quatérnios. Por fim, são aplicadas as expressões 52, 53 e 54 para o cálculo do ângulo *yaw*. Considere-se  $s \equiv \sin$  e  $c \equiv \cos$ . Os  $180^\circ$  são adicionados no fim do cálculo para se obter ângulos entre  $0^\circ$  e  $360^\circ$ :

$$M_x = MagN_x * c(Pitch) + MagN_z * s(Pitch) \quad (52)$$

$$M_y = MagN_x s(Roll) s(Pitch) + MagN_y c(Roll) - MagN_z s(Roll) c(Pitch) \quad (53)$$

$$yaw = 180 + \tan^{-1} \left( \frac{M_y}{M_x} \right) \quad (54)$$

Seguem-se os diferentes filtros aplicados para obter a orientação e posição do veículo.

### 5.3.1. Filtro Complementar

Como já referido, o **filtro complementar**, é um filtro não probabilístico que relaciona duas ou mais variáveis com o intuito de obter a melhor estimativa possível sobre uma determinada variável. Deste modo, o fluxograma da Figura 48 resume o filtro complementar implementado.



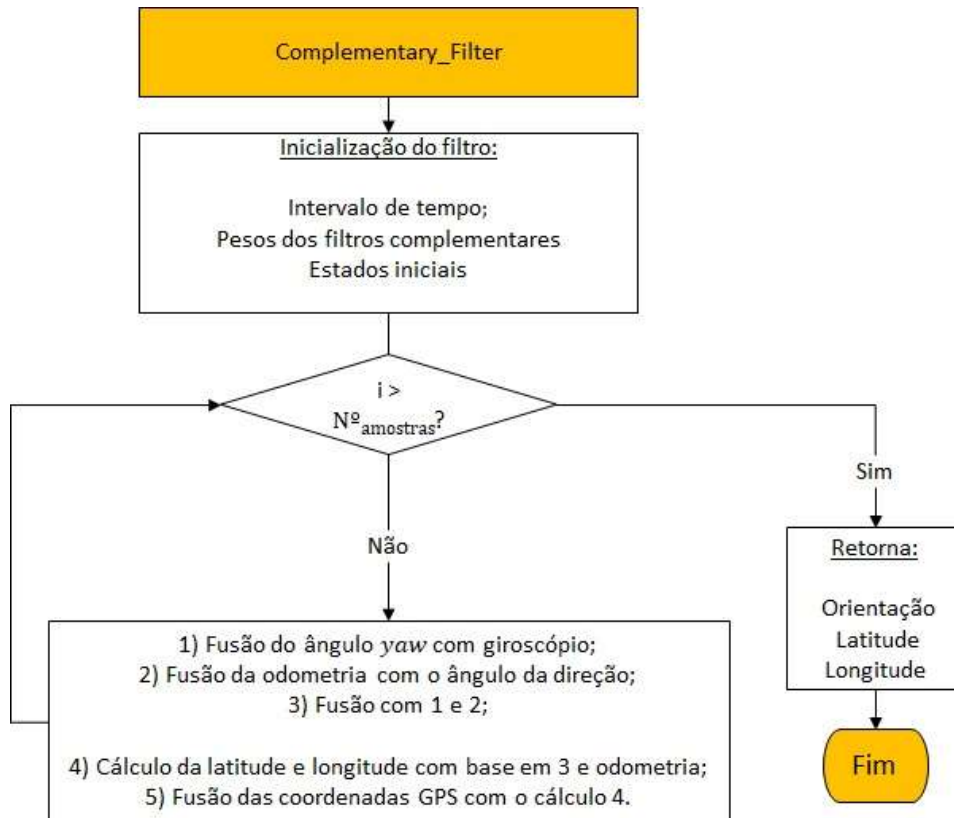


Figura 48 - Fluxograma referente à função Filtro Complementar.

De forma a tornar a implementação mais estruturada, a função encontra-se dividida entre o cálculo da orientação e o cálculo da latitude e longitude final. No entanto, o cálculo da posição encontra-se dependente do cálculo da orientação. Posto isto, são pré-definidas todas as variáveis necessárias para os diferentes filtros, tais como, intervalo de tempo, peso, valores iniciais do filtro, características do veículo (perímetro da roda) e características do odómetro (número máximo de pulsos por volta). Sempre que o número da amostra for inferior às amostras totais, o ciclo repete-se, iniciando-se com a fusão do ângulo *yaw* com o giroscópio. A expressão 55 descreve o filtro 1) aplicado onde,  $\alpha_1 = 0.95$  representa o peso do filtro,  $h = 0.025$  o intervalo de tempo,  $Fused_{yaw}(k) = yaw(1)$  o valor inicial do filtro e  $vang_z$  representa os valores medidos pelo giroscópio no eixo -z.

$$Fused_{yaw}(k) = \alpha_1 * (Fused_{yaw}(k - 1) + vang_z(k) * h) + (1 - \alpha_1) * yaw(k - 1) \quad (55)$$

Em seguida realiza-se a determinação da orientação baseada unicamente na odometria e no valor do ângulo da direção do veículo. Para isso, é necessário utilizar a expressão 35 e integrá-la com o tempo. A expressão 56 traduz o ciclo implementado:

$$R_{yaw}(k) = R_{yaw}(k-1) + h \left( \frac{P_A(k) * D_{pp}}{L} \right) \tan(\gamma_D(k)) \quad (56)$$

onde  $R_{yaw}$  é a orientação relativa calculada,  $h$  o intervalo de tempo,  $P_A$  o número de pulsos gerados pelo alternador num intervalo de tempo,  $D_{pp}$  a distância em metros a que um pulso corresponde,  $L$  a distância entre as rodas do veículo e  $\gamma_D$  o ângulo da direção do veículo. Para a dissertação os valores utilizados foram  $D_r = 1.125$  metros e  $D_{pp} = 0.087$  metros (expressão 40). Por fim, o  $R_{yaw}$  e o  $Fused_{yaw}$  são fundidos para se obter a orientação final ( $\theta$ ). O peso para cada valor foi definido como  $\alpha_3 = 0.5$ . Na equação 57 pode-se ver o cálculo efetuado.

$$\theta(k) = \alpha_3 * Fused_{yaw}(k) + (1 - \alpha_3) * R_{yaw}(k) \quad (57)$$

Para a determinação da latitude e longitude foram aplicadas as expressões 42 e 43, ambas integradas com o tempo. A expressão 58 e 59 traduzem o cálculo efetuado onde  $y_{rel}$  e  $x_{rel}$  representam o cálculo da latitude e longitude relativas, respetivamente, levando em consideração a última posição e a odometria;  $MtoD$  representa a conversão de metros para graus. De acordo com [81] e considerando a latitude e longitude onde a dissertação foi realizada,  $MtoD = 12,70 * 10^{-6}$ :

$$y_{rel}(k) = y_{rel}(k-1) + (h * P_A(k) * D_{pp} * \cos(\theta(k)) * MtoD) \quad (58)$$

$$x_{rel}(k) = x_{rel}(k-1) + (h * P_A(k) * D_{pp} * \sin(\theta(k)) * MtoD) \quad (59)$$

Por fim, realiza-se um filtro complementar para fundir os valores medidos pelo GPS com os valores calculados. A expressão 60 e 61 traduzem esse cálculo onde  $\alpha_5 = 0.7$ :

$$y(k) = \alpha_5 * GPS_{Lat} + (1 - \alpha_5) * y_{rel}(k) \quad (60)$$

$$x(k) = \alpha_5 * GPS_{Lng} + (1 - \alpha_5) * x_{rel}(k) \quad (61)$$

### 5.3.2. Filtro de Kalman Extended

O **filtro de Kalman Extended** é um estimador ótimo que leva em consideração não só as medições, mas também previsões realizadas através do modelo do sistema. Como referido, o filtro trabalha numa base de predição-atualização recursiva e o fluxograma da Figura 49 resume a implementação do mesmo.

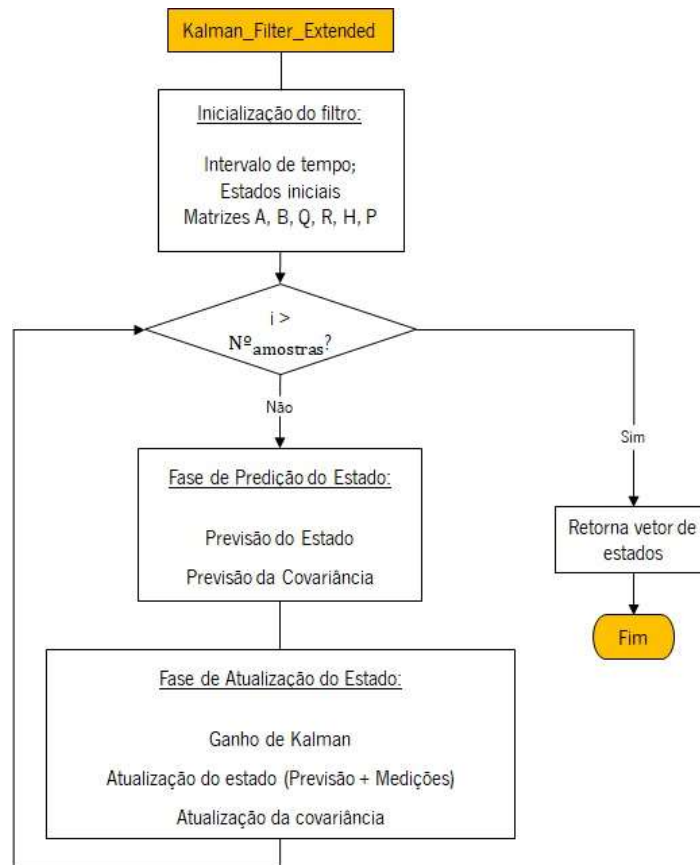


Figura 49 - Fluxograma referente à função Kalman\_Filter\_Extended.

A função inicia-se com a definição do intervalo de tempo, os estados iniciais do filtro (necessários para a primeira iteração deste) e as matrizes que o constituem. O modelo de estados define matematicamente a evolução do sistema ao longo do tempo através das suas equações de estado. São, por isso, utilizadas no momento da previsão do filtro e quanto melhor traduzirem o

sistema menor será o erro associado à sua previsão. O vetor de estados considerado é composto por cinco variáveis,

$$VetorEstados = [x, y, v, \theta, \omega]^T \quad (62)$$

Sendo que,  $x$  representa a posição relativamente ao eixo -x (longitude),  $y$  representa a posição relativamente ao eixo -y (latitude),  $v$  representa a velocidade linear,  $\theta$  a orientação e  $\omega$  a velocidade angular do veículo. As equações que descrevem o movimento do veículo em tempo discreto encontram-se apresentadas nas equações 63, 64, 65, 66 e 67 [82]:

$$x_k = x_{k-1} + hv_{k-1} \cos(\theta_{k-1}) + \frac{1}{2}h^2 ax_{k-1} \cos(\theta_{k-1}) \quad (63)$$

$$y_k = y_{k-1} + hv_{k-1} \sin(\theta_{k-1}) + \frac{1}{2}h^2 ay_{k-1} \sin(\theta_{k-1}) \quad (64)$$

$$v_k = (P_A(k-1) * D_{pp})/h \quad (65)$$

$$\theta_k = \theta_{k-1} + \omega_{k-1} * h \quad (66)$$

$$\omega_k = \frac{v_{k-1}}{L} \tan(\gamma_{k-1}) \quad (67)$$

Sendo que,  $x, y, v, ax, ay, \theta, \omega$  correspondem à posição relativamente ao eixo -x, à posição relativamente ao eixo -y, à velocidade linear, aceleração relativamente ao eixo -x, aceleração relativamente ao eixo -y, à orientação e velocidade angular do veículo, respetivamente;  $P_A$  o número de pulsos gerados pelo alternador;  $D_{pp}$  a distância em metros a que cada pulso corresponde;  $L$  a distância entre a roda da frente e a roda de trás do veículo e  $\gamma$  o ângulo da direção do veículo.

De forma a se obter uma melhor estimativa, o filtro utiliza valores de covariância que tentam emular os erros presentes no sistema, isto é, possíveis discrepâncias do valor esperado. Se o erro associado ao modelo de estados for muito elevado então o filtro tenderá a confiar mais nas medições e vice-versa. Este erro nunca pode ser modelado como zero porque levaria o filtro a nunca atualizar. Deste modo, assume-se que a covariância entre variáveis é nula e a matriz  $Q$  é dada pela matriz da expressão 68:

$$Q = \text{diagonal}[\sigma_x^2 \ \sigma_y^2 \ \sigma_v^2 \ \sigma_\theta^2 \ \sigma_\omega^2]^T \quad (68)$$

Uma vez que o valor do erro do processo é algo extremamente complicado de definir com exatidão, muitas aplicações definem estes valores aleatoriamente ou a posteriori. No entanto, para este tipo de aplicações, e de acordo com [83][84], foram utilizados os seguintes valores:

$$\sigma_x = \frac{dt^2 * ax_{Max}}{2} \quad (69)$$

$$\sigma_y = \frac{dt^2 * ay_{Max}}{2} \quad (70)$$

$$\sigma_v = dt * a_{Max} \quad (71)$$

$$\sigma_\theta = \frac{dt^2 * \alpha_{Max}}{2} \quad (72)$$

$$\sigma_\omega = dt * \alpha_{Max} \quad (73)$$

Relativamente ao modelo das medições, este é constituído por todas as medições disponíveis para fundir com os dados preditos no modelo de estados. Neste caso, a matriz de medições é constituída por:

$$z_k = [X_{GPS}, Y_{GPS}, V_{GPS}, yaw, 0]^T \quad (74)$$

Sendo que,  $X_{GPS}$ ,  $Y_{GPS}$  e  $V_{GPS}$  correspondem às medições realizadas pelo GPS e  $yaw$  ao valor calculado pelo magnetómetro compensado pelo acelerómetro. O número de medições pode provir de múltiplas fontes desde que estas sejam medições do estado, contudo, para os efeitos desta dissertação, foram utilizadas as medições do GPS, do magnetómetro e do acelerómetro. Por fim, a matriz H define quais estados são atribuídas a cada medição, neste caso, é uma matriz identidade.

Por fim, a incerteza presente nas medições é semelhante à matriz Q. Estes erros fornecerão ao filtro a capacidade de ponderar sobre em quem confiar (medições ou previsões). Estes valores são geralmente obtidos através de testes ou fornecidos pelos fabricantes dos sensores, não existindo uma fórmula geral para os determinar. Por exemplo, o GPS é um caso particular dado que, dependendo do número de satélites em vista ou das condições atmosféricas, o erro associado

à medição deste pode variar. Através da *string* NMEA, este é capaz de fornecer uma ideia de qual será a precisão de uma leitura. Este parâmetro é denominado de HDOP (*Horizontal Dilution Of Precision*) e  $HDOP \in [0, 20[$  e quanto maior o seu valor pior a exatidão do sinal.

### 5.3.3. Filtro de Kalman Unscented

O filtro de **Kalman Unscented**, como já referido, é uma variação do filtro de Kalman capaz de lidar com sistemas não lineares recorrendo a uma abordagem estatística, tendo por base a transformação *unscented*. Sendo um filtro de Kalman, funciona numa base de predição-atualização. Recorrendo ao último estado, é retirado um número pré-definido de amostras (o número varia consoante o número de estados). Essas amostras são propagadas pelo modelo de estados e através das medições. Após a propagação de ambas, é calculado o peso relativo a cada uma das propagações e os dados são fundidos de forma a se obter uma solução ponderada. As equações 18 a 34 resumem as equações do filtro, e a Figura 50 ilustra o algoritmo implementado.

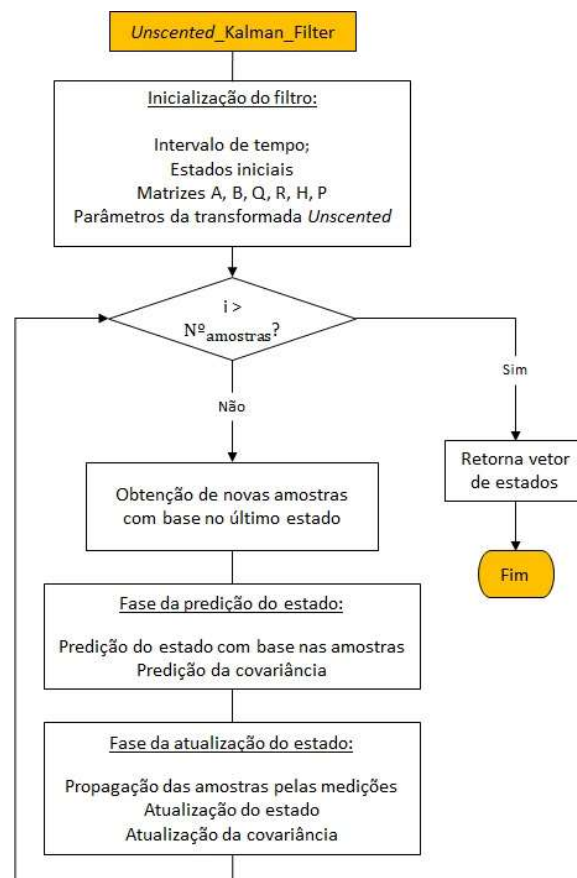


Figura 50 - Fluxograma referente à função *Unscented\_Kalman\_Filter*.

À semelhança da função *Kalman\_Filter\_Extended*, a função *Unscented\_Kalman\_Filter* inicia-se com a definição de parâmetros iniciais como o intervalo de tempo, estados iniciais do filtro (necessários para a primeira iteração deste), matrizes que o constituem e os parâmetros de dimensionamento da transformada *Unscented*. Relativamente aos parâmetros de dimensionamento da transformada *Unscented* foram utilizados os seguintes valores:

- $\alpha = 1$ ;
- $\beta = 2$ ;
- $k = 0$ ;
- $\lambda = \alpha^2(L + k) - L$ , onde  $L$  representa o número de estados considerados.

Da mesma forma que no filtro de Kalman, as expressões 63, 64, 65, 66 e 67 descrevem o movimento do veículo. Relativamente à incerteza do modelo (matriz  $Q$ ) e à incerteza das medições (matriz  $R$ ) foram utilizadas as mesmas matrizes para permitir a comparação entre os dois filtros.

#### 5.4. Interface Gráfica

A **interface gráfica** do sistema é responsável por mostrar os resultados obtidos em gráficos e num mapa de forma a facilitar a interpretação destes. A aplicação foi desenvolvida em CSS, HTML e JavaScript e sempre que executada lê os vetores de dados e desenha no mapa as diferentes trajetórias. O HTML é a linguagem padrão para a criação de páginas e aplicações *web* e é responsável pela parte estática da interface. O CSS é um mecanismo ou uma ferramenta que permite adicionar ou alterar cores, fontes, espaçamentos, entre outras coisas, sendo por isso responsável pela parte estética da interface. Por fim, o JavaScript é uma linguagem de alto nível e é caracterizada por ser dinâmica e multi-paradigma. Desta forma, permite uma programação imperativa, funcional e com recurso a eventos, tornando possível a criação de aplicações e páginas *web* interativas. Em conjunto, o HTML, CSS e JavaScript constituem as tecnologias fundamentais da *World Wide Web (WWW)*.

De forma a representar as diferentes trajetórias no mapa foi utilizado a API (*Application Programming Interface*) gratuita do *Google Maps*. De acordo com [85], qualquer representação realizada no *Google Maps* encontra-se afetada por um erro quadrático médio de 1.59m. Esse erro pode variar uma vez que a API gratuita não utiliza imagens satélite atualizadas.

A interface gráfica contém apenas o mapa com as diferentes trajetórias e a identificação do autor, universidade, curso, empresa e ano de realização. A Figura 51 ilustra o que pode ser encontrado na aplicação.



Figura 51 - Interface gráfica do sistema de localização.





## 6. Resultados Experimentais

Após o desenvolvimento do trabalho, descrito no capítulo 5, foram realizados múltiplos testes ao sistema. Neste capítulo são demonstrados e analisados três desses testes, uma vez que são exemplos do tipo de testes realizados. É importante realçar que todos eles foram realizados com a bicicleta da Figura 44 e, de forma a determinar a distância e representar os percursos realizados foi utilizado o *Google Maps*.

### 6.1. Teste 1

O teste 1 consiste num percurso retilíneo com 111 m e teve uma duração de 44 s (Figura 52.a). De acordo com a odometria do veículo registou-se uma distância percorrida de 107.81 m e uma velocidade média de 2.45 m/s. Deste modo, e utilizando o *Google Maps* como referência, obteve-se um erro de 3.19 m por parte da odometria. Relativamente à leitura do GPS, no momento inicial, obteve-se um erro de 5 m, tendo sido rapidamente corrigido com o movimento do veículo. Analisando a Figura 52.b) e Figura 53 é visível uma ponderação entre a trajetória descrita pela odometria e a trajetória descrita pelo GPS sendo perceptível uma suavização da mesma. Comparando o resultado dos três filtros aplicados não se verifica diferenças significativas.

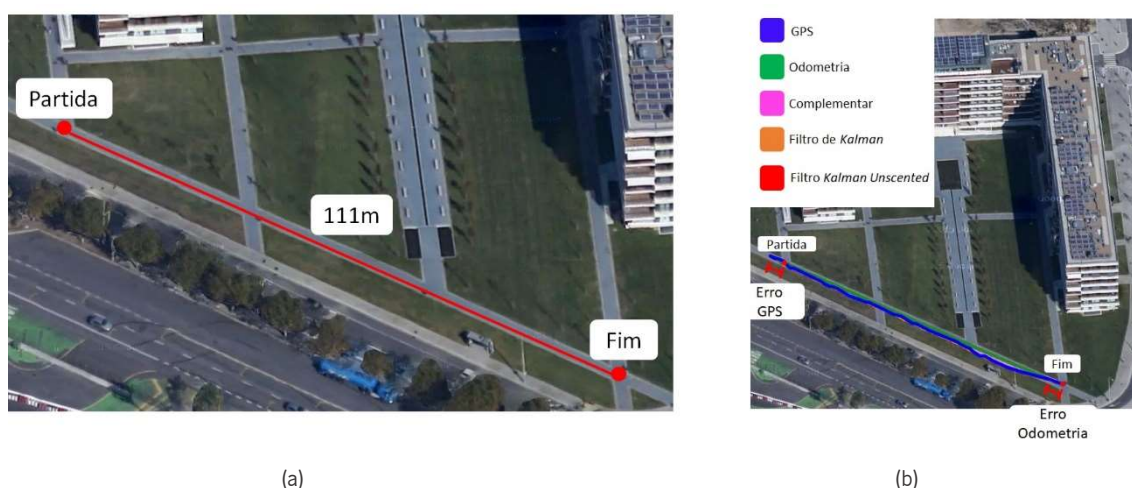


Figura 52 - a) Percurso realizado no Teste 1 evidenciando a distância, ponto de partida e chegada. b) Diferentes trajetórias medidas e calculadas pelos diferentes sensores e filtros, respetivamente.



Figura 53 – Aumento da parte final da figura 53.b).

Na Figura 54 encontra-se a medição da orientação do veículo durante o teste 1. Aqui, é possível visualizar os valores antes da fusão sensorial (magenta) e depois da fusão sensorial (restantes cores). É visível que existe uma suavização dos valores da medição da orientação a magenta, isto é: uma redução de ruído quando o veículo se encontra em repouso, passando de oscilações de  $\pm 1.2^\circ$  para oscilações menores que  $0.2^\circ$ ; uma redução no ruído quando o veículo se encontra em movimento, passando de oscilações de  $\pm 3.2^\circ$  para oscilações menores que  $0.5^\circ$ ; redução de *outliers*, passando de picos de  $7.5^\circ$  para picos com valores menores que  $1^\circ$ .

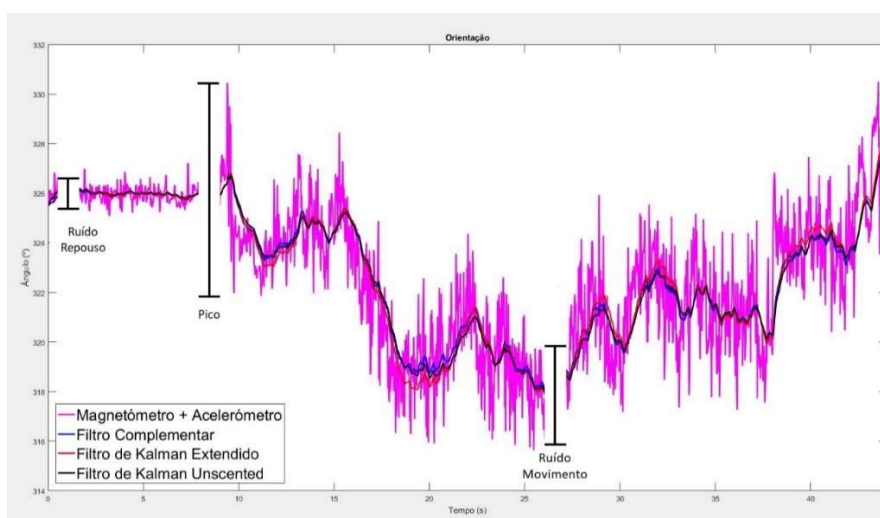


Figura 54 - Orientação do veículo durante o Teste 1.

## 6.2. Teste 2

O teste 2 consiste num percurso com 180 m e teve uma duração de 36 s (Figura 55.a). De acordo com a odometria registou-se uma distância percorrida de 174.37 m e uma velocidade média de 4.81 m/s. Assim, obteve-se um erro proveniente da odometria de 5.63 m. Relativamente à leitura do GPS, registou-se um erro máximo de 5 m. Analisando a Figura 55.b), verifica-se novamente uma suavização da trajetória. Contudo, quando ocorre a mudança de direção verifica-se uma trajetória exagerada por parte do GPS e o contrário por parte da odometria. Deste modo, os filtros aplicados fornecem uma média entre ambas, aproximando-se da trajetória real. Na parte final, verifica-se um desvio da trajetória real tanto por parte do GPS como pela odometria. É visível que nenhum dos filtros foi capaz de estimar corretamente o ponto de chegada, contudo, os resultados dos filtros de Kalman foram os que mais se aproximaram da trajetória real, não demonstrando diferenças significativas entre si.

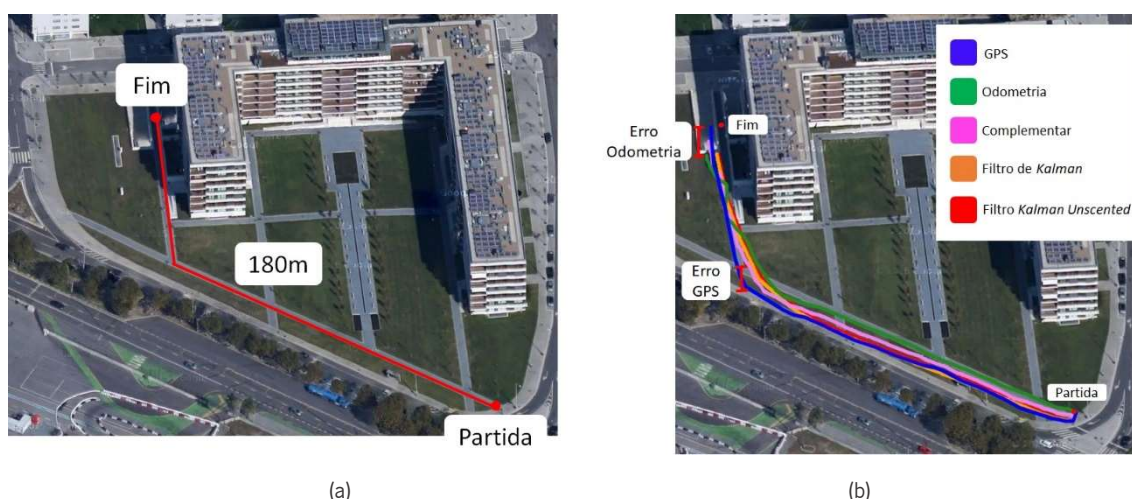


Figura 55 - a) Percurso realizado no Teste 2 evidenciando a distância, ponto de partida e chegada. b) Diferentes trajetórias medidas e calculadas pelos diferentes sensores e filtros, respetivamente.

Na Figura 56 encontra-se a medição da orientação do veículo durante o teste 2 e, como no caso anterior, é possível visualizar e comparar o valor da orientação calculada com (magenta) e sem (outras cores) fusão sensorial. É visível que existe uma suavização dos valores da medição da orientação a magenta, isto é: uma redução de ruído quando o veículo se encontra em repouso, passando de oscilações de  $\pm 3^\circ$  para oscilações menores que  $0.2^\circ$ ; uma redução no ruído quando o veículo se encontra em movimento, passando de oscilações de  $\pm 27^\circ$  para oscilações inferiores

a 2 °. Por fim, é visível a ocorrência de um *outlier* de 342 ° que, após os filtros, passa a ter um valor inferior a 22 °.

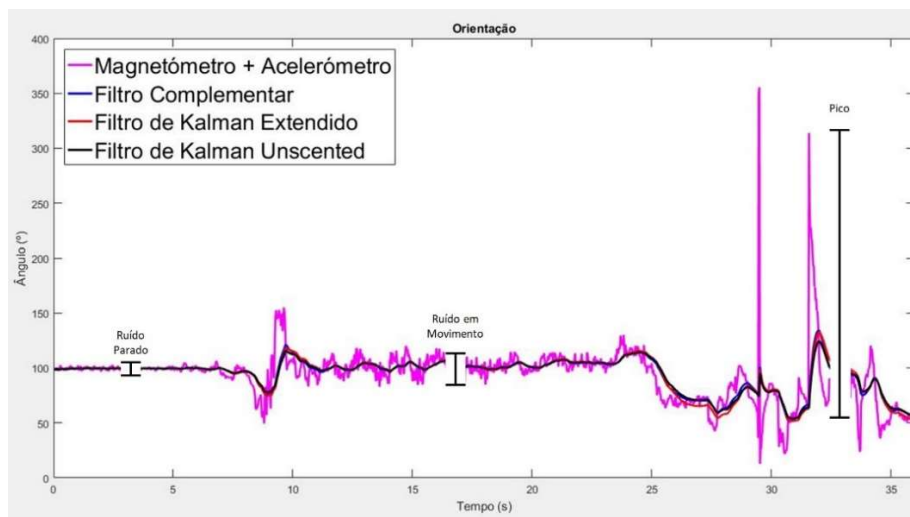


Figura 56 - Orientação do veículo durante o teste 2.

### 6.3. Teste 3

O teste 3 consiste num percurso mais complexo de 520 m, constituído por troços retilíneos e troços em curva e teve uma duração de 88 s (Figura 57.a). De acordo com a odometria registou-se uma distância percorrida de 453.7 m e uma velocidade média de 5.19 m/s. Deste modo, obtém-se um erro aproximado de 66.m por parte da odometria. Analisando a Figura 57.b) verifica-se que o GPS possui um erro inicial e uma trajetória exagerada quando se realizam curvas. Na parte final, o erro do GPS atinge o valor máximo de 10 m. A odometria, por outro lado, apresenta uma acumulação de um erro por defeito, indicando que na parte final se atravessou o edifício. Para efeitos de teste e para verificar a eficiência da introdução do HDOP na atribuição do valor das covariâncias, aplicou-se uma variação desta quando o valor do GPS atingiu o valor máximo. Assim, na curva final, o valor incerteza da medição foi aumentado significativamente de forma a confiar muito na previsão, resultando numa correção acentuada do erro proveniente do GPS. Em suma, os filtros aplicados fornecem uma estimativa ponderada entre ambas partes, originando um resultado mais próximo da trajetória real, no entanto, ainda com algumas falhas. Entre todos os filtros, o complementar foi o que apresentou o pior resultado. Os resultados obtidos com os diferentes filtros de Kalman não apresentaram diferenças significativas entre si.

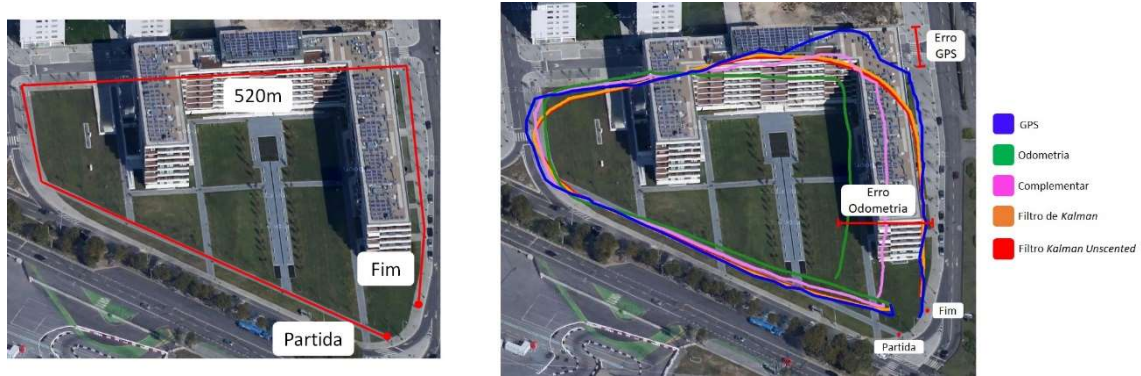


Figura 57 - a) Percurso realizado no Teste3 evidenciando a distância, ponto de partida e ponto de chegada; b) Diferentes trajetórias medidas e calculadas pelos diferentes sensores e filtros, respetivamente.

Na Figura 58 encontra-se a medição da orientação do veículo durante o teste 3 e, da mesma forma, é possível visualizar e comparar o valor da orientação com (magenta) e sem (outras cores) fusão sensorial. Deste modo, é visível que existe uma redução de ruído quando o veículo se encontra em repouso, passando de oscilações de  $\pm 3.7^\circ$  para oscilações inferiores a  $0.2^\circ$ ; quando o veículo se encontra em movimento passa-se de oscilações de  $\pm 12.5^\circ$  para oscilações de  $\pm 0.5^\circ$ ; e a redução de picos, por exemplo, redução de um pico de  $55^\circ$  para  $3^\circ$ .

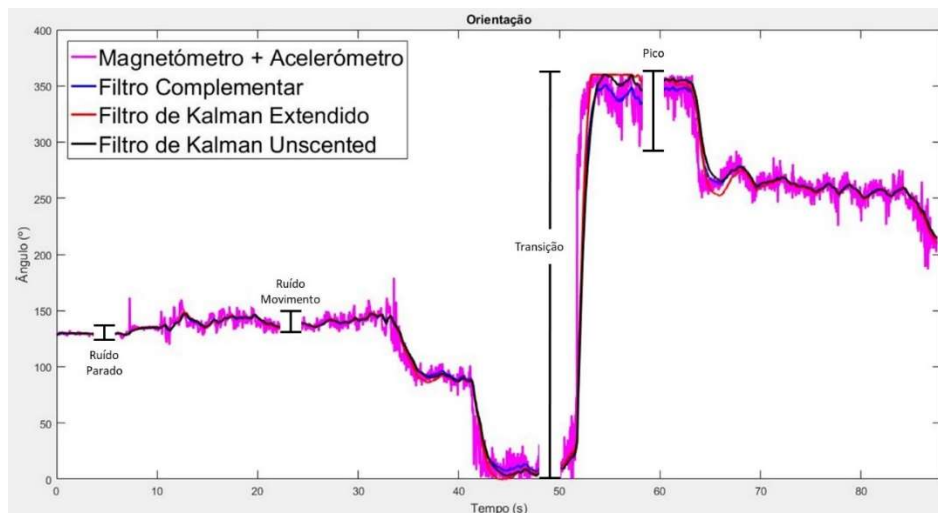


Figura 58 - Orientação do veículo durante o Teste 3.

#### 6.4. Análise comparativa dos testes

Os três testes apresentam diferentes graus de complexidade de forma a perceber o funcionamento do sistema. Deste modo, a Tabela 4 resume as principais características dos diferentes testes.



## Resultados Experimentais

Tabela 4 - Resumo dos testes realizados.

	Teste 1	Teste 2	Teste 3
<b>Distância</b>	111 m	180 m	520 m
<b>Velocidade Odometria</b>	2.45 m/s	4.81 m/s	5.19 m/s
<b>Erro Odometria</b>	3.19 m	5.63 m	66.3 m
<b>Erro Máximo GPS</b>	5 m	5 m	10 m
<b>Ruído Repouso Orientação</b>	<0.2 °	<0.2 °	<0.2 °
<b>Ruído Movimento Orientação</b>	<0.5 °	<2 °	<0.5 °

Analisando a Tabela 4 verifica-se que quanto maior a distância do teste maior o erro associado à odometria. Verifica-se também que, quanto maior a complexidade do trajeto realizado e com odómetro implementado, o filtro complementar não consegue realizar uma ponderação sempre correta. Relativamente à introdução de uma variação das covariâncias com base no HDOP, verifica-se um resultado satisfatório por parte dos filtros de Kalman. Relativamente ao erro proveniente do GPS, o teste 3, teve o maior entre os três testes dado que foi o teste com maior proximidade ao edifício. Relativamente aos filtros aplicados, o filtro complementar mostrou ser bastante eficiente em trajetórias simples. Contudo, quando as trajetórias se tornam mais complexas, os filtros de Kalman produzem melhores resultados. Não se verificaram diferenças significativas entre os resultados produzidos pelos diferentes tipos de filtro de Kalman.

Quanto à medição da orientação, todos os filtros apresentaram notórias melhorias na redução do ruído presente no sistema não apresentando diferenças significativas. O filtro de Kalman *Unscented* revelou-se o mais estável após grandes variações de valores de orientação (360 ° para 0 ° ou vice-versa, por exemplo).

Por fim, relativamente à comunicação de dados, ambas as soluções exploradas cumprem com os requisitos necessários. No entanto, o cartão  $\mu SD$  não oferece uma comunicação em tempo real e o módulo XBee utilizado encontra-se limitado pelo alcance máximo de 1000m.

## 7. Conclusões e trabalho futuro

Neste capítulo apresentam-se as conclusões relativas ao trabalho desenvolvido e algumas sugestões de trabalho futuro.

Esta dissertação foi realizada em parceria com o CEiiA, que pretende inovar na mobilidade com o desenvolvimento de um veículo autónomo. Os veículos autónomos são um tema cada vez mais recorrente devido à sua aplicabilidade. No entanto, a condução autónoma é um tema complexo e envolve diversas áreas, entre as quais a autolocalização do veículo. Conseguir-lo de uma forma robusta, eficiente e económica reveste-se de particular importância.

O objetivo principal desta dissertação consistiu no desenvolvimento de um sistema de autolocalização para um veículo autónomo. Para tal, foi necessário o desenvolvimento de três unidades capazes de recolher, processar e mostrar os resultados obtidos. Para a primeira unidade (URD), desenvolveu-se uma placa de circuito impresso que integra o módulo sensorial (um recetor GPS, uma IMU 9-DOF, um *encoder* absoluto e um alternador convertido num *encoder* incremental), o módulo de alimentação, o módulo de comunicação (um cartão  $\mu SD$  e um módulo XBee S2C Pro), o microcontrolador (Atmega328p) e ainda todo o *firmware* necessário ao funcionamento do microcontrolador. A segunda unidade (UPD) foi desenvolvida utilizando o *software Matlab*, onde se implementaram filtros matemáticos (o filtro Complementar, o filtro de *Kalman Extended* e o filtro de *Kalman Unscented*). A terceira unidade consiste numa interface gráfica que permite a visualização dos resultados e foi desenvolvida recorrendo a HTML, CSS e *Javascript*.

Relativamente aos resultados obtidos na medição da orientação do veículo, numa fase inicial verificaram-se oscilações de  $\pm 3^\circ$  em repouso e  $\pm 15^\circ$  em movimento. Após a aplicação dos filtros obtiveram-se oscilações inferiores a  $\pm 0.2^\circ$  em repouso e inferiores a  $\pm 2^\circ$  em movimento. É constatável também a mitigação de *outliers* presentes no sistema, obtendo-se no final um sinal mais estável. O ruído presente na medição da orientação pode provir de erros de medição ou de perturbações inerentes à irregularidade do piso ou à natural instabilidade do veículo utilizado para testes (bicicleta).

Quanto à medição da posição do veículo, em trajetórias simples todos os filtros apresentam um bom desempenho. No entanto, com o aumento da complexidade das trajetórias e da distância



percorrida, nota-se uma perda de eficiência por parte do filtro complementar. O GPS apresentou um erro máximo de 10 m em situações em que os sinais provenientes dos satélites se encontravam parcialmente bloqueados por um edifício. A odometria mostrou ser satisfatória, apresentando um erro de 12.75 % para a trajetória mais longa e complexa. Não se registaram diferenças significativas entre os resultados obtidos com os diferentes filtros de *Kalman* implementados.

Relativamente ao trabalho futuro, seguem-se algumas sugestões para melhorar alguns aspetos do sistema desenvolvido:

- Desenvolvimento de uma nova placa de circuito impresso de forma a reduzir custos e dimensões;
- Adaptação do sistema desenvolvido a uma plataforma móvel tradicional (automóvel);
- Exploração de um módulo de comunicação com maior alcance;
- Adaptação da interface gráfica para ambiente *online*.

# Referências

- [1] R. Siegwart and I. R. Nourbakhsh, "Introduction to Autonomous Mobile Robots," *MIT Press*, 2004.
- [2] R. Vivacqua, R. Vassallo, and F. Martins, "A low cost sensors approach for accurate vehicle localization and autonomous driving application," *Sensors (Switzerland)*, vol. 17, no. 10, pp. 1–33, Oct. 2017.
- [3] J. J. Leonard and H. F. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," *IEEE Trans. Robot. Autom.*, vol. 7, no. 3, pp. 376–382, 1991.
- [4] J. M. Esteves, "Metodologia de Autolocalização Absoluta em Ambientes Quase-Estruturados," Universidade do Minho, 2005.
- [5] J. Borenstein, H. R. Everett, and L. Feng, "Where am I? Sensors and methods for mobile robot positioning," *Univ. Michigan*, 1996.
- [6] M. Drumheller, "Technical Report - Mobile Robot Localization Using Sonar," *Massachusetts Inst. Technol. - Artif. Intell. Lab.*, pp. 1–24, 1985.
- [7] J. S. Esteves, A. Carvalho, and C. Couto, "Generalized geometric triangulation algorithm for mobile robot absolute self-localization," in *IEEE International Symposium on Industrial Electronics*, 2003, vol. I, pp. 346–351.
- [8] S. Se, D. Lowe, and J. Little, "Mobile Robot Localization and Mapping with Uncertainty using Scale-Invariant Visual Landmarks," *Int. J. Rob. Res.*, vol. 21, no. 8, pp. 735–758, 2002.
- [9] D. Ronzoni, R. Olmi, C. Secchi, and C. Fantuzzi, "AGV global localization using indistinguishable artificial landmarks," *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 287–292, 2011.
- [10] E. Menegatti, A. Pretto, A. Scarpa, and E. Pagello, "Omnidirectional Vision Scan Matching for Robot Localization in Dynamic Environments," *IEEE Trans. Robot.*, vol. 22, no. 3, 2006.
- [11] B. Barshan and H. F. Durrant-Whyte, "Inertial Navigation Systems for Mobile Robots," *IEEE Trans. Robot. Autom.*, vol. 11, no. 3, pp. 328–342, 1995.

- [12] H. Li, F. Nashashibi, and G. Toulminet, "Localization for intelligent vehicle by fusing mono-camera, low-cost GPS and map data," *IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC*, no. October 2010, pp. 1657–1662, 2010.
- [13] J. R. Lowell, "Military applications of localization, tracking, and targeting," *IEEE Wirel. Commun.*, vol. 18, no. 2, pp. 60–65, 2011.
- [14] R. Anacleto, L. Figueiredo, A. Almeida, and P. Novais, "Localization system for pedestrians based on sensor and information fusion," *17th Int. Conf. Inf. Fusion (FUSION). IEEE. 2014*, vol. 59, no. 11, p. 8, 2014.
- [15] X. Han, H. J. Kim, C. W. Jeon, H. C. Moon, and J. H. Kim, "Development of a low-cost GPS/INS integrated system for tractor automatic navigation," *Int. J. Agric. Biol. Eng.*, vol. 10, no. 2, pp. 123–131, 2017.
- [16] E. Guizzo, "How Google's Self-Driving Car Works - IEEE Spectrum," 2011. [Online]. Available: <https://spectrum.ieee.org/autaton/robotics/artificial-intelligence/how-google-self-driving-car-works>. [Accessed: 09-Nov-2017].
- [17] "SPAN® GNSS/INS Combined Systems | High-Precision GPS Devices | High-Precision GNSS Devices | Products | NovAtel." [Online]. Available: <https://www.novatel.com/products/span-gnss-inertial-systems/span-combined-systems/>. [Accessed: 20-Nov-2017].
- [18] "MTi-G-710 - Products - Xsens 3D motion tracking." [Online]. Available: <https://www.xsens.com/products/mti-g-710/>. [Accessed: 04-Sep-2018].
- [19] H. Martinez-Barber and D. Herrero-Pierez, "Autonomous navigation of an automated guided vehicle in industrial environments," *Robot. Comput. Integr. Manuf.*, vol. 26, no. 4, pp. 296–311, 2010.
- [20] J.-T. Li and H. Liu, "Design Optimization of Amazon Robotics," *Autom. Control Intell. Syst.*, vol. 4, no. 2, p. 48, 2016.
- [21] "CEiiA." [Online]. Available: <https://www.ceiia.com/>. [Accessed: 12-Dec-2017].
- [22] J. Byun, M. Roh, K. I. Na, J. C. Sohn, and S. Kim, "Navigation and localization for autonomous vehicle at road intersections with low-cost sensors," in *Lecture Notes in*

- Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7508 LNAI, no. PART 3, Springer, Berlin, Heidelberg, 2012, pp. 577–587.
- [23] A. Nemra and N. Aouf, “Robust INS/GPS sensor fusion for UAV localization using SDRE nonlinear filtering,” *IEEE Sens. J.*, vol. 10, no. 4, pp. 789–798, 2010.
  - [24] J. Kim, J. Cheng, J. Guivant, and J. Nieto, “Compressed fusion of GNSS and inertial navigation with simultaneous localization and mapping,” *IEEE Aerosp. Electron. Syst. Mag.*, vol. 32, no. 8, pp. 22–36, 2017.
  - [25] S. Sukkarieh, E. M. Nebot, and H. F. Durrant-Whyte, “A High Integrity IMU/GPS Navigation Loop for Autonomous Land Vehicle Applications,” *IEEE Trans. Robot. Autom. Int. J. Robot. Res. Int. J. Robot. Res.*, vol. 151314, no. 4, 1999.
  - [26] M. Andrejasic, “MEMS Accelerometers,” University of Ljubljana, 2008.
  - [27] C. Reinholtz *et al.*, “Odin: Team VictorTango’s entry in the DARPA Urban Challenge,” *Springer Tracts Adv. Robot.*, vol. 56, no. 8, pp. 125–162, 2009.
  - [28] S. Panzieri, F. Pascucci, and G. Ulivi, “An outdoor navigation system using GPS and inertial platform,” *IEEE/ASME Trans. Mechatronics*, vol. 7, no. 2, pp. 134–142, Jun. 2002.
  - [29] C. Li, Me. Zhang, and K. Liu, “Unmanned ground vehicle positioning system by GPS/dead-reckoning/IMU sensor fusion,” *Adv. Eng. Res. - 2nd Annu. Int. Conf. Electron.*, vol. 117, no. Eeeis 2016, pp. 737–747, 2017.
  - [30] M. Ibne, “Indoor and Outdoor Localization of a Mobile Robot Fusing,” Northeastern University, 2017.
  - [31] E. North, J. Georgy, U. Iqbal, M. Tarbochi, and A. Noureldi, “Improved Inertial/Odometry/GPS Positioning of Wheeled Robots Even in GPS-Denied Environments,” *Glob. Navig. Satell. Syst. Signal, Theory Appl.*, no. 11, pp. 257–278, 2012.
  - [32] M. Spangenberg, V. Calmettes, and J. Y. Tournet, “Fusion of GPS, INS and odometric data for automotive navigation,” *Eur. Signal Process. Conf.*, pp. 886–890, 2007.
  - [33] S. Wang, Z. Deng, and G. Yin, “An accurate GPS-IMU/DR data fusion method for driverless car based on a set of predictive models and grid constraints,” *Sensors (Switzerland)*, vol.

- 16, no. 3, pp. 1–13, 2016.
- [34] Bourns, “Bourns EMS22A Absolute Encoder Datasheet,” 2012.
- [35] C. Arantes, “Desenvolvimento de uma plataforma móvel omnidirecional baseada em rodas Mecanum com gestão otimizada da energia,” Universidade do Minho, 2014.
- [36] J. Gu, M. Meng, A. Cook, and P. X. Liu, “Sensor fusion in mobile robot: some perspectives,” *Proc. 4th World Congr. Intell. Control Autom. (Cat. No.02EX527)*, vol. 2, no. February, pp. 1194–1199, 2002.
- [37] J. Borenstein and L. Feng, “Gyrodometry: A New Method for Combining Data from Gyros and Odometry in Mobile Robots,” *IEEE Int. Conf. Robot. Autom.*, pp. 423–428, 1996.
- [38] J. M. Calvário and J. M. Vasconcelos, “Estratégias de Fusão Sensorial para Sistemas de Navegação com Aplicação a Helicópteros Autónomos,” Instituto Superior Técnico, 2002.
- [39] C. Fauvel, “Avoid Gimbal Lock for Rotation/Direction Maya Manipulators,” *Blog: around the corner*, 2012. [Online]. Available: <http://around-the-corner.typepad.com/adn/2012/08/avoid-gimbal-lock-for-rotationdirection-maya-manipulators.html>. [Accessed: 26-Jun-2018].
- [40] S. Marques, “Small Satellites Attitude Determination Methods,” Instituto Superior Técnico, 2000.
- [41] D. Gebre-Egziabher, G. H. Elkaim, J. D. Powell, and B. W. Parkinson, “A Gyro-Free Quaternion-Based Attitude Determination System Suitable for Implementation Using Low Cost Sensors,” *IEEE Position Locat. Navig. Symp.*, p. 8, 2000.
- [42] A. Kim and M. F. Golnaraghi, “A quaternion-based orientation estimation algorithm using an inertial measurement unit,” *PLANS 2004. Position Locat. Navig. Symp. (IEEE Cat. No.04CH37556)*, no. December, pp. 268–272, 2004.
- [43] A. Bueno and R. Romano, “Filtro complementar aplicado a medida de inclinação de plataformas móveis,” *Vasa*, p. 6, 2011.
- [44] M. Euston, P. Coote, R. Mahony, J. Kim, and T. Hamel, “A complementary filter for attitude estimation of a fixed-wing UAV,” *2008 IEEE/RSJ Int. Conf. Intell. Robot. Syst. IROS*, pp. 340–345, 2008.

- [45] T. Islam, M. S. Islam, M. Shajid-UI-Mahmud, and M. Hossam-E-Haider, "Comparison of complementary and Kalman filter based data fusion for attitude heading reference system," *AIP Conf. Proc.*, vol. 1919, p. 10, 2017.
- [46] D. G. Macharet, A. A. Neto, V. Costa, M. Fernando, and M. Campos, "Mobile Robot Localization in Outdoor Environments Using Complementary Filtering," *ABCM Symp. Ser. Mechatronics*, vol. 4, pp. 758–767, 2010.
- [47] A. Ricardo and T. Pessanha, "Sistema móvel de telemetria para automóveis," Instituto Superior de Engenharia do Porto, 2013.
- [48] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," *UNC-Chapel Hill, TR 95-041*, vol. 7, no. 1, pp. 1–16, 2006.
- [49] M. B. Rhudy, R. A. Salguero, and K. Holappa, "A Kalman Filtering Tutorial for Undergraduate Students," *Int. J. Comput. Sci. Eng. Surv.*, vol. 08, no. 01, pp. 01-18, 2017.
- [50] A. M. Santana, A. A. S. Souza, R. S. Britto, P. J. Alsina, and A. A. D. Medeiros, "Localization of a Mobile Robot based in Odometry and Natural Landmarks using Extended Kalman Filter.," *ICINCO 2008, Proc. Fifth Int. Conf. Informatics Control. Autom. Robot.*, no. January, p. 7, 2008.
- [51] A. Skobeleva, V. Ugrinovskii, and I. Petersen, "Extended Kalman Filter for indoor and outdoor localization of a wheeled mobile robot," *2016 Aust. Control Conf. AuCC 2016*, pp. 212–216, 2017.
- [52] D. Simon, "Using Nonlinear Kalman Filtering to Estimate Signals," *IEEE Trans. Automat. Contr.*, vol. 13, no. 1, pp. 83–86, 2013.
- [53] M. Rhudy and Y. Gu, "Understanding Nonlinear Kalman Filters, Part I: Selection between EKF and UKF," *Interact. Robot. Lett.*, vol. 149, no. 3, pp. 713–728, 2013.
- [54] M. Rhudy and Y. Gu, "Understanding Nonlinear Kalman Filters, Part II: An Implementation Guide," *Interact. Robot. Lett.*, no. June, pp. 1–18, 2013.
- [55] S. J. Julier and J. K. Uhlmann, "Unscented Filtering and Nonlinear Estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, 2004.
- [56] Matlab, "Understanding Kalman Filters, part 5: Nonlinear State Estimators Video -

- MATLAB.” [Online]. Available: <https://www.mathworks.com/videos/understanding-kalman-filters-part-5-nonlinear-state-estimators-1495052905460.html>. [Accessed: 21-Sep-2018].
- [57] M. Persson and A. Håkansson, “A communication protocol for different communication technologies in cyber-physical systems,” *Procedia Comput. Sci.*, vol. 60, no. 1, pp. 1697–1706, 2015.
- [58] “Serial Peripheral Interface (SPI) - learn.sparkfun.com.” [Online]. Available: <https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi>. [Accessed: 08-Apr-2018].
- [59] Y. Y. Fang and X. J. Chen, “Design and simulation of UART serial communication module based on VHDL,” *2011 3rd Int. Work. Intell. Syst. Appl. ISA 2011 - Proc.*, vol. 8, p. 4, 2011.
- [60] “Serial Communication - learn.sparkfun.com.” [Online]. Available: <https://learn.sparkfun.com/tutorials/serial-communication/all>. [Accessed: 08-Apr-2018].
- [61] F. Leens, *An Introduction To I2C and SPI Protocols*. Institute of Electrical and Electronics Engineers, 2009.
- [62] “I2C - learn.sparkfun.com.” [Online]. Available: <https://learn.sparkfun.com/tutorials/i2c>. [Accessed: 08-Apr-2018].
- [63] “RoHS 2 - Electronics waste - Environment - European Commission.” [Online]. Available: [http://ec.europa.eu/environment/waste/rohs\\_eee/legis\\_en.htm](http://ec.europa.eu/environment/waste/rohs_eee/legis_en.htm). [Accessed: 06-Sep-2018].
- [64] Quectel, “L80 Hardware Design GPS Module Series Datasheet,” 2014.
- [65] “Marine Electronics | Boating Electronics | About National Marine Electronics Association.” [Online]. Available: [https://www.nmea.org/content/about\\_the\\_nmea/about\\_the\\_nmea.asp](https://www.nmea.org/content/about_the_nmea/about_the_nmea.asp). [Accessed: 08-Apr-2018].
- [66] “TinyGPS++ | Arduiniana.” [Online]. Available: <http://arduiniana.org/libraries/tinygpsplus/>. [Accessed: 10-Sep-2018].
- [67] “Bi-Directional Logic Level Converter Hookup Guide - learn.sparkfun.com.” [Online].

- Available: [https://learn.sparkfun.com/tutorials/bi-directional-logic-level-converter-hookup-guide?\\_ga=2.248576071.1375128782.1523197127-1779024967.1517243242](https://learn.sparkfun.com/tutorials/bi-directional-logic-level-converter-hookup-guide?_ga=2.248576071.1375128782.1523197127-1779024967.1517243242).  
[Accessed: 08-Apr-2018].
- [68] Adafruit, “Adafruit 9-DOF Absolute Orientation IMU Fusion Breakout - BNO055.” [Online]. Available: <https://www.adafruit.com/products/2472%5Cninternal-pdf://0.0.3.4/2472.html>. [Accessed: 11-Sep-2018].
- [69] Bosch Sensortec, “Datasheet - BNO055 Intelligent 9-axis absolute orientation sensor,” no. November, p. 105, 2014.
- [70] C. Barz and C. Oprea, “Study of Electromagnetic Field in Claw-Poles,” *Ann. Fac. Eng. Hunedoara - Int. J. Eng.*, pp. 359–362, 2011.
- [71] “Alternator,” 2013. [Online]. Available: <https://en.wikipedia.org/wiki/Alternator>.
- [72] H. Roller and B. Compatible, “Nexus Hub Dynamo,” *SHIMANO Datasheet*, p. 6000.
- [73] V. Semiconductors, “Datasheet - Vishay Semiconductors Optocoupler, Phototransistor Output, Low Input Current,” 2018.
- [74] V. Technology, “Datasheet for: Industrial SDHC microSD Card,” 2017.
- [75] Digi, “XBee®/XBee-PRO S2C Zigbee® RF Module User Guide,” 2018.
- [76] D. W. Hart, “Power Electronics,” *Princ. Electron.*, pp. 577–600, 1950.
- [77] R. Pepy, A. Lambert, and H. Mounier, “Path Planning using a Dynamic Vehicle Model,” *2006 2nd Int. Conf. Inf. Commun. Technol.*, vol. 1, pp. 781–786, 2006.
- [78] B. Paden, M. Cap, S. Z. Yong, D. Yershov, and E. Frazzoli, “A Survey of Motion Planning and Control Techniques for Self-driving Urban Vehicles,” *IEEE Trans. Intell. Veh.*, vol. 56, p. 27, 2016.
- [79] J. A. Farrell, “Aided Navigation: GPS with High Rate Sensors,” *CEUR Workshop Proc.*, vol. 1542, pp. 33–36, 2008.
- [80] L. Tran, “Data Fusion with 9 Degrees of Freedom Inertial Measurement Unit To Determine Object’s Orientation,” California Polytechnic State University, 2017.
- [81] Wikipedia, “Decimal degrees,” 2015. [Online]. Available:

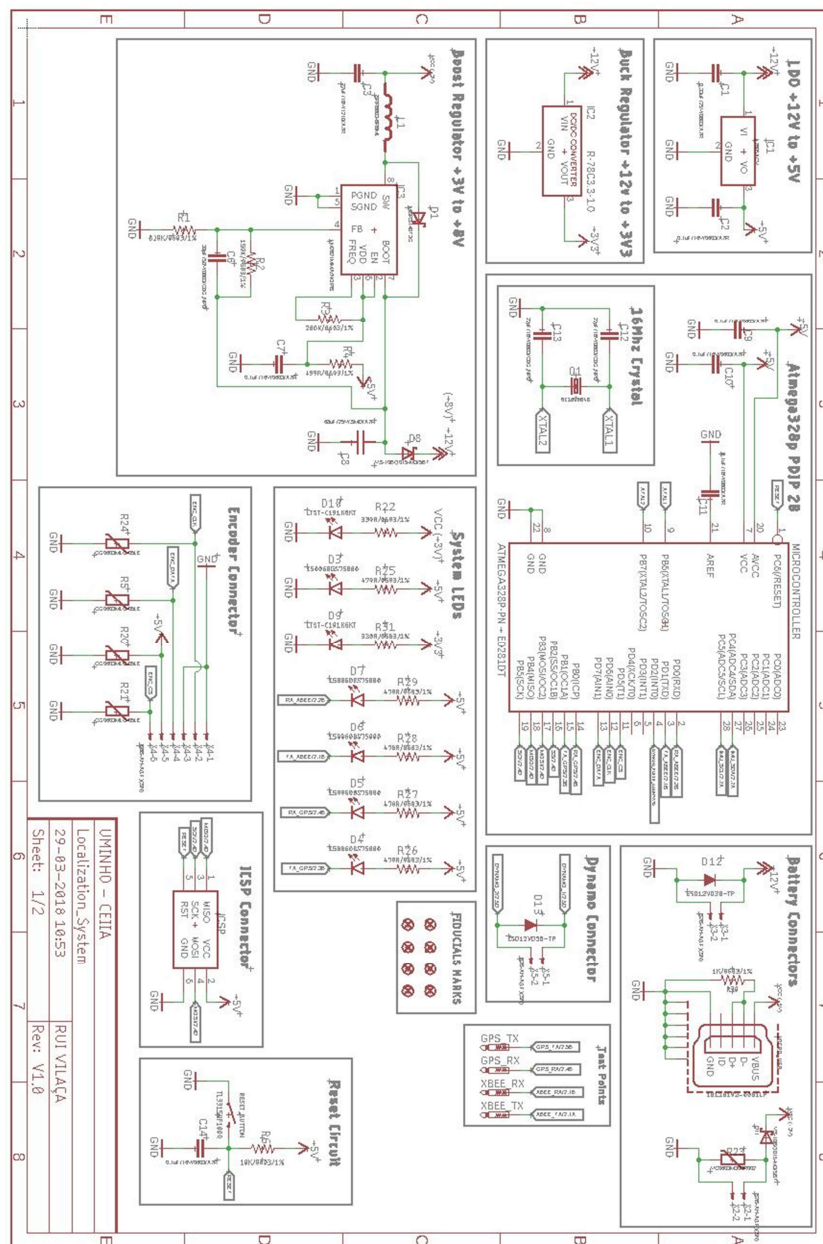


- [https://en.wikipedia.org/wiki/Decimal\\_degrees](https://en.wikipedia.org/wiki/Decimal_degrees). [Accessed: 21-Sep-2018].
- [82] S. Baek, C. Liu, P. Watta, and Y. L. Murphey, "Accurate vehicle position estimation using a Kalman filter and neural network-based approach," *2017 IEEE Symp. Ser. Comput. Intell. SSCI 2017 - Proc.*, vol. 2018–Janua, no. August, pp. 1–8, 2018.
- [83] A. Kelly, "A 3D state space formulation of a navigation Kalman filter for autonomous vehicles," The Robotics Institute of Carnegie Mellon University, 1994.
- [84] P. H. King and C. F. Reinholtz, "A Low Cost Localization Solution Using a Kalman Filter for Data Fusion," Virginia Tech, 2008.
- [85] N. Z. Mohammed, A. Ghazi, and H. E. Mustafa, "Positional accuracy testing of Google Earth," *Int. J. Multidiscip. Sci. Eng.*, vol. 4, no. 6, pp. 6–9, 2013.
- [86] "Data Transmission - Parallel vs Serial Transmission." [Online]. Available: <https://www.quantil.com/content-delivery-insights/content-acceleration/data-transmission/>. [Accessed: 10-Sep-2018].
- [87] W. Nelson *et al.*, "Use of Circulr Error Probability in Target Detection," *United States Air Force - Electron. Syst. Div.*, 1988.

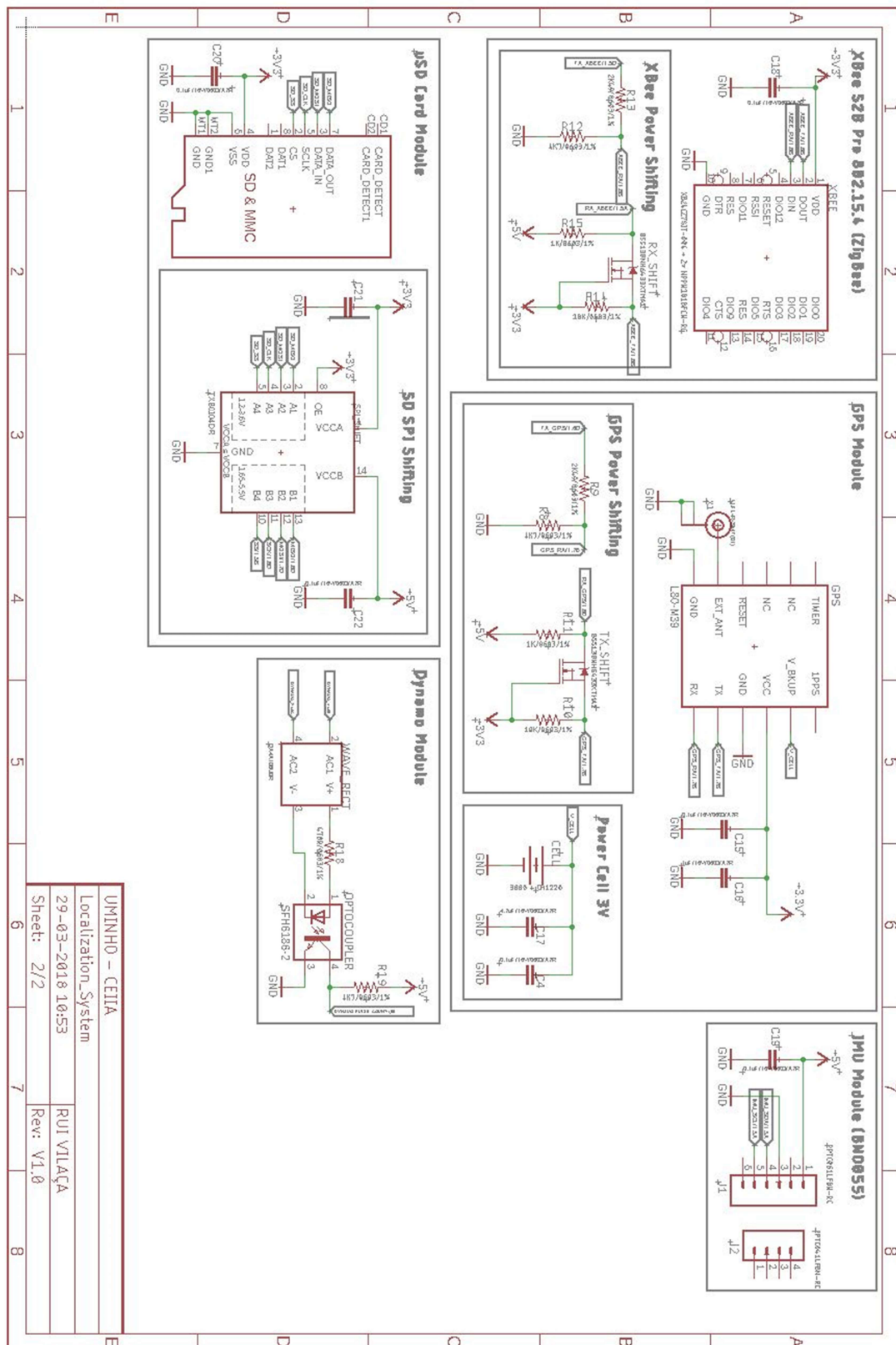
# Apêndices

## Apêndice A

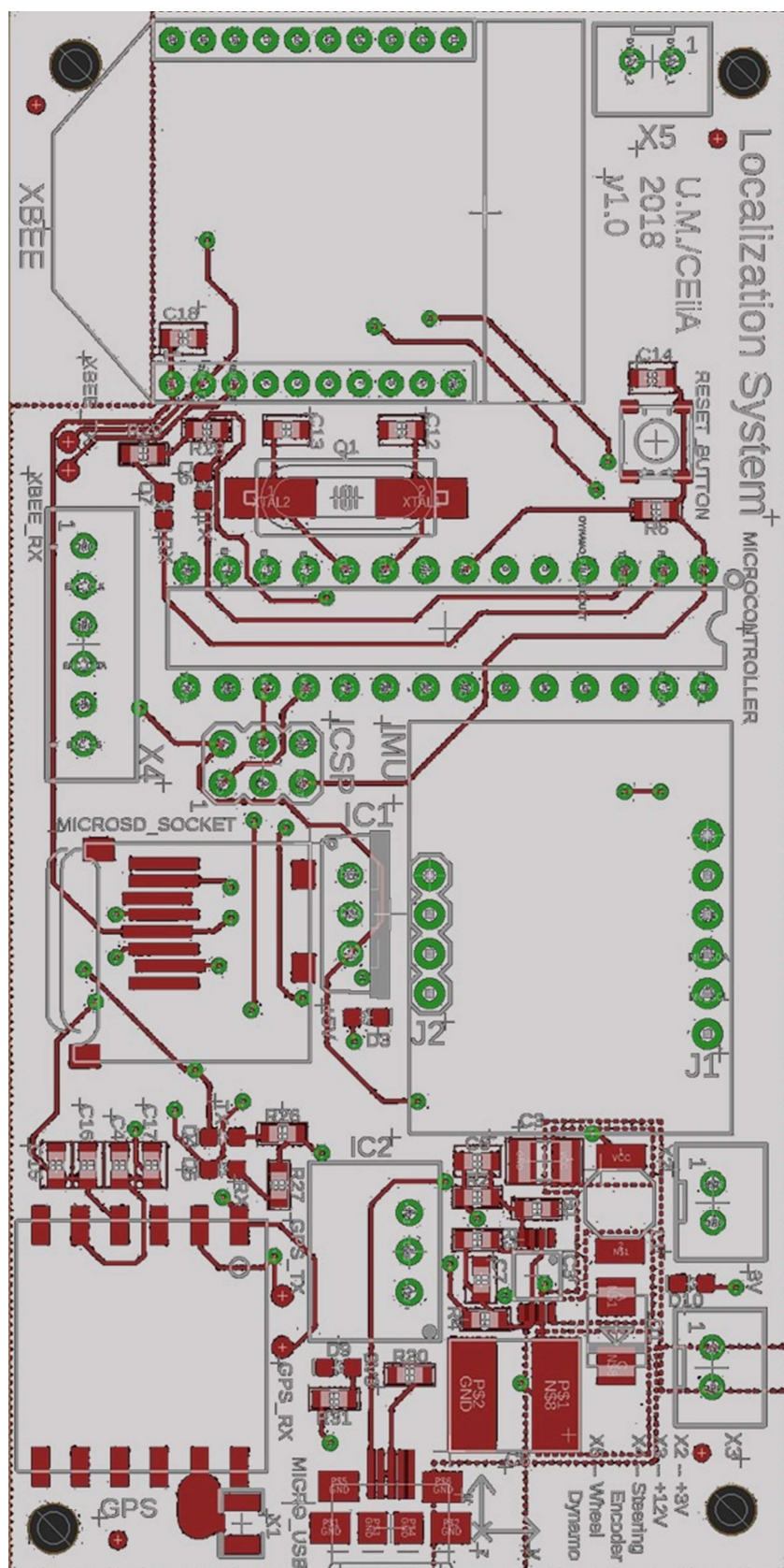
Aqui encontram-se o esquemático (A. 1 e A. 2) e o *layout* (A. 3 e A. 4) completo da placa de circuito impressa desenvolvida.



A. 1 - Primeira parte do esquemático da placa de circuito impresso desenvolvida.

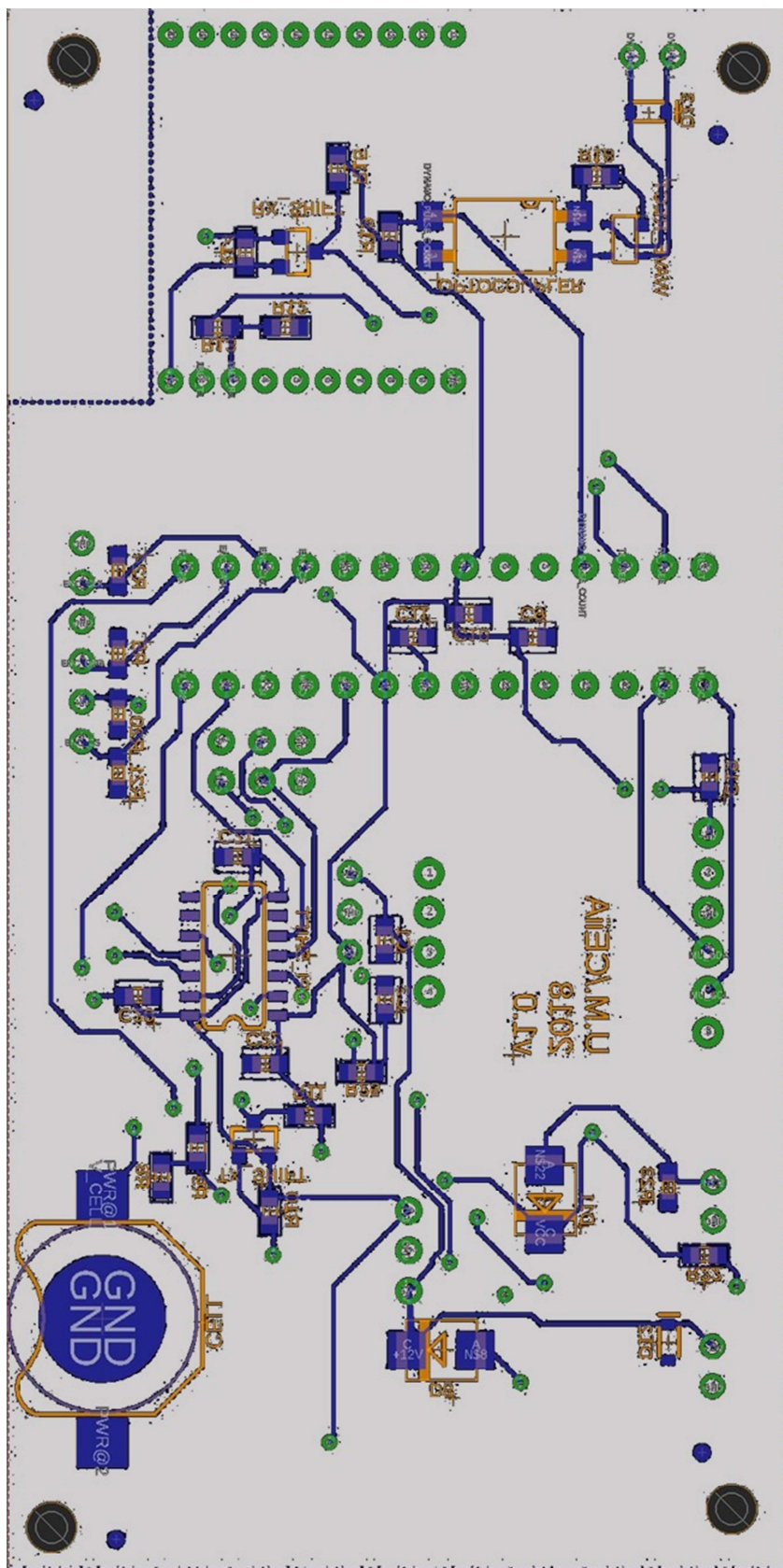


A. 2 - Segunda parte do esquemático da placa de circuito impressa desenvolvida.



A. 3 - Layout referente à parte superior da placa de circuito impressa desenvolvida.





A. 4 - *Layout* referente à parte inferior da placa de circuito impressa desenvolvida.